

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

**Modelos Proativos
para
Hiperídia Adaptativa**

por

LUIZ ANTÔNIO MORO PALAZZO

Tese submetida à avaliação, como requisito
parcial para a obtenção do grau de
Doutor em Ciência da Computação

Prof. Dr. José Mauro Volkmer de Castilho
Orientador

Prof. Dr. Antônio Carlos da Rocha Costa
Co-orientador



UFRGS

SABi



85230137

Porto Alegre, janeiro de 2000

UFRGS
INSTITUTO DE INFORMÁTICA
BIBLIOTECA

Catálogo na Publicação

Palazzo, Luiz Antônio Moro

Modelos Proativos para Hipermídia Adaptativa / por Luiz Antônio Moro Palazzo. - Porto Alegre: PPGC da UFRGS, 2000.

114 p.: il.

Tese (doutorado) - Universidade Federal do Rio Grande do Sul. Programa de Pós Graduação em Computação. Porto Alegre, 2000. Orientador: Castilho, José Mauro Volkmer de. Co-orientador: Costa, Antônio Carlos da Rocha.

1.Hipermídia Adaptativa. 2.Modelos Proativos. 3.World Wide Web. 4.Teoria do Fecho. 5.Lógica das Situações. 6.Sistemas Educacionais Adaptativos. I.Castilho, José Mauro Volkmer de. II.Costa, Antônio Carlos da Rocha. III. Modelos Proativos para Hipermídia Adaptativa.

| INSTITUTO DE INFORMÁTICA BIBLIOTECA | | |
|--|-----------|---------|
| N.º CHAMADA: 581304111111 P155m | N.º REG.: | 37285 |
| ORIGEM: ① | 31,00,00 | 7 30,00 |
| FUNDO: II | FORN.: | II |

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
 Reitora: Profa. Wrana Panizzi
 Pró-Reitor de Graduação: Prof. Franz Rainer Semmelmann
 Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux
 Coordenadora do PPGC: Profa. Carla Maria Dal Sasso Freitas
 Bibliotecária-Chefe do Instituto de Informática: Beatriz Haro

Agradecimentos

Esta tese representa a concretização de um trabalho de vários anos, ao longo dos quais acumularam-se agradecimentos devidos aqueles que, no labor incessante e no convívio diário, compartilharam os muitos momentos da sua elaboração. O autor agradece, com risco de omissão involuntária:

Postumamente, ao Prof. José Mauro Volkmer de Castilho, grande amigo, conselheiro e mestre, a cuja memória esta tese é dedicada.

Ao grande amigo Prof. Antônio Carlos da Rocha Costa que, com sabedoria e experiência, tão bem soube orientar o desenvolvimento e a conclusão deste trabalho.

Aos professores do Programa de Pós Graduação em Computação do Instituto de Informática da UFRGS, pela reconhecida dedicação, conhecimento e competência, que tornam esta universidade um centro de excelência em computação no país.

Aos colegas e amigos do Grupo de Pesquisa em Bancos de Dados Inteligentes, no Instituto de Informática da UFRGS: Mara, Rafael, Miguel, Fabiane, Gisele, Carine, Rodrigo, Álvaro, Leandro, Maurício e Juliano.

Aos colegas e amigos da Escola de Informática da UCPel: Paulo Ricardo, Mário, Caruso, Beti, Adenauer, Jorge, Cristiano, Aroldo, Graçaliz, Luiz Fernando, João Avelino, Regina, Paula, Boaventura, Cava, Gláucius, Paulinho, Xico, Clara, Cláudio, Simone, Lúcia, Adéle, Nescau, Fábio, André, Luis Alessandro, Ramiro, Eduardo, Stanley, Mauro, Gladmir, Edécio, Ane, Christiano, Mônica e Rúbia.

Aos colegas e amigos da Faculdade de Educação da UFPel: Ceres, Solange, Marcos, Teca, Magda, Márcia, Elomar, Colvara, Alessandro e Rita.

Aos colegas e amigos do Centro de Informática da UFPel: Ricardo, Adenauer, Cláudio, Alexandre, Tuca, Sandokan, Sônia, Vera, Teresa, Heloísa, Cristina, Lúcio, Bayard, Roger, Henrique e Luis.

Aos colegas e amigos do Grupo de Pesquisa em Inteligência Artificial da UCPel: Robson, Lisane, Maurício, Ana Cláudia, Augusto, Maria Helena, Luciana, Maria Luísa, Rogério, Márcio, Antônio e Alexandre.

À UCPel, UFPel, UFRGS, PUCRS, FURG, ESIN, FaE, EDUCAT, CNPq, FAPERGS e CAPES,

À minha mãe, Santina e meus irmãos Sérgio e Carlos.

À minha esposa, Mara e minhas filhas Daniela e Renata.

A todos, pelo amor, carinho, amizade, dedicação e afeto, pelo ensinamento, pela compreensão, estímulo e incentivo, pelo apoio, confiança e fé, o meu mais profundo e sincero reconhecimento.

Esta tese é dedicada à memória do
Prof. José Mauro Volkmer de Castilho

Sumário

| | |
|---|----|
| Lista de Abreviaturas | 8 |
| Lista de Símbolos | 9 |
| Lista de Figuras | 10 |
| Lista de Tabelas | 11 |
| Resumo | 12 |
| Abstract | 13 |
| 1. Introdução | 14 |
| 1.1 Números | 14 |
| 1.2 Hipermídia Adaptativa | 15 |
| 1.3 Modelos de Adaptação na Web | 16 |
| 1.4 Objetivos da Pesquisa | 17 |
| 1.5 Organização da Tese | 17 |
| 2. Modelos de Hipertexto e a World Wide Web | 18 |
| 2.1 Hipertexto e Hipermídia | 18 |
| 2.2 Modelos para Sistemas de Hipertexto | 20 |
| 2.2.1 HAM ou Hypertext Abstract Machine | 20 |
| 2.2.2 O Modelo Trellis | 21 |
| 2.2.3 O modelo Dexter | 22 |
| 2.2.4 Um Modelo Formal de Hipertexto | 23 |
| 2.2.5 O Modelo Torre | 24 |
| 2.3 Revisão Técnica da World Wide Web | 24 |
| 2.3.1 Arquitetura Cliente-Servidor | 25 |
| 2.3.2 A Web como um Sistema Hipertexto | 26 |
| 2.4 A WWW e os Modelos Clássicos de Hipertexto | 28 |
| 3. Hipermídia Adaptativa | 30 |
| 3.1 Conceitos Básicos | 30 |
| 3.2 Modelagem do Usuário | 32 |
| 3.3 Métodos e Técnicas de Adaptação | 34 |
| 3.3.1 Métodos de Apresentação Adaptativa | 34 |
| 3.3.2 Técnicas de Apresentação Adaptativa | 35 |
| 3.3.3 Métodos de Navegação Adaptativa | 37 |
| 3.3.4 Técnicas de Navegação Adaptativa | 39 |
| 4. Um Modelo Conexcionista para Hipermídia Adaptativa | 43 |
| 4.1 Introdução | 43 |
| 4.2 Redes de Informação | 44 |
| 4.3 Inferência em Redes através de Operadores de Fecho | 45 |
| 4.4 Adaptação Proativa na Web | 47 |
| 4.5 Parâmetros de Adaptação | 48 |
| 4.6 O Processo de Adaptação | 51 |

| | |
|--|----|
| 5. Um Modelo de Adaptação Simbólica | 53 |
| 5.1 Introdução à Teoria das Situações | 53 |
| 5.2 Relevância | 54 |
| 5.3 Combinação de Informações | 55 |
| 5.3.1 Fusão | 55 |
| 5.3.2 Composição | 56 |
| 5.3.3 Degradação | 56 |
| 5.3.4 Contenção | 56 |
| 5.3.5 Preclusão | 57 |
| 5.3.6 Ínfons Parametrizados | 57 |
| 5.4 Ínfons, Documentos e Situações | 58 |
| 5.4.1 Tipos | 59 |
| 5.5 Combinação de Situações | 59 |
| 5.5.1 União | 59 |
| 5.5.2 Fusão | 60 |
| 5.5.3 Composição | 60 |
| 5.5.4 A Relação “é sobre” entre Documentos | 61 |
| 5.6 Raciocínio sobre Situações | 62 |
| 5.6.1 Postulados Básicos | 63 |
| 5.6.2 Postulados de Combinação | 63 |
| 5.6.3 Postulados Baseados em Ínfons | 65 |
| 5.6.4 Negação | 66 |
| 5.7 Redes Situadas | 66 |
| 5.8 Sobreposição de Modelos | 69 |
| 6. Uma Arquitetura para HA Proativa | 71 |
| 6.1 Introdução | 71 |
| 6.2 Uma Arquitetura Genérica | 71 |
| 6.3 A Interface do Usuário | 74 |
| 6.4 O Processo de Modelagem | 75 |
| 6.5 A Base de Modelos de Usuários | 76 |
| 6.6 O Processo de Adaptação | 77 |
| 6.7 Especificação Lógica do Sistema | 79 |
| 7. ia@net : Um Sistema Educacional Proativo | 81 |
| 7.1 Sistemas Educacionais Adaptativos na Web | 81 |
| 7.2 ia@net: Uma Visão Geral | 82 |
| 7.3 Estruturas de Modelagem Física | 83 |
| 7.4 O Modelo Pedagógico | 85 |
| 7.4.1 O Modelo Educacional | 85 |
| 7.4.2 A Fonte de Hipermissão | 86 |
| 7.4.3 A Estrutura de um Módulo | 88 |
| 7.5 Apresentação do Sistema e o Processo de Login | 91 |
| 7.6 Modelagem do Estudante | 91 |
| 7.7 Implementação do Processo de Adaptação | 94 |
| 8. Conclusões e Perspectivas de Trabalhos Futuros | 96 |
| 8.1 Modelos Proativos para Hipermissão Adaptativa | 96 |

| | |
|--|------------|
| 8.1.1 O Componente Conexionista | 97 |
| 8.1.2 O Componente Simbólico | 98 |
| 8.1.3 Integração Dos Componentes | 98 |
| 8.2 Uma Arquitetura para HA Proativa | 99 |
| 8.3 ia@net: Adaptação Proativa na Web | 99 |
| 8.4 Perspectivas de Trabalhos Futuros | 100 |
| Anexo 1 | 101 |
| Anexo 2 | 104 |
| Bibliografia | 108 |

Lista de Abreviaturas

| | |
|---------|--|
| AA | Anotação Adaptativa |
| ACM | Association for Computing Machinery |
| AIMA | Artificial Intelligence: A Modern Approach |
| bme | Base de modelos de estudantes |
| bmu | Base de modelos de usuários |
| CA | Classificação Adaptativa |
| CERN | Laboratório Europeu de Física das Partículas |
| CF | Classificação de Fragmentos |
| CG | Classificação Global |
| CGI | Common Gateway Interface |
| CL | Condução Local |
| EA | Explicação Adicional |
| EC | Explicação Comparativa |
| EV | Explicação Variante |
| Fapergs | Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul |
| FAQ | Frequently Asked Questions |
| FTP | File Transfer Protocol |
| FV | Fragmento Variante |
| HA | Hipermídia Adaptativa |
| HAM | Hypertext Abstract Machine |
| HTTP | Hypertext Transfer Protocol |
| IA | Inteligência Artificial |
| II | Interfaces Inteligentes |
| MA | Mapas Adaptativos |
| ME | Modelo do Estudante |
| MU | Modelo do usuário |
| NAPI | Núcleo de Pesquisa e Desenvolvimento em Informática |
| OD | Orientação Direta |
| OL | Orientação Local |
| OG | Orientação Global |
| PLN | Processamento da Linguagem Natural |
| PV | Página Variante |
| rs | Rede Situada |
| rf | Rede Fechada |
| SEA | Sistemas Educacionais Adaptativos |
| SH | Sistemas de Hipermídia |
| SHA | Sistemas de Hipermídia Adaptativa |
| STI | Sistemas Tutores Inteligentes |
| UCPel | Universidade Católica de Pelotas |
| URI | Uniform Resource Identifier |
| URL | Uniform Resource Locator |
| URN | Uniform Resource Name |
| VDM | Vienna Development Method |
| WAIS | Wide Area Information System |
| WWW | World Wide Web |

Lista de Símbolos

| | |
|-------------------|-----------------------------|
| + | Adição |
| \wedge | Composição de ínfons |
| \cap | Composição de situações |
| K_f | Constante de frequência |
| K_s | Constante de simetria |
| K_T | Constante de transitividade |
| \rightarrow | Contenção |
| \vee | Degradação |
| $\ll \gg$ | Delimitadores |
| \downarrow | É sobre |
| \Leftrightarrow | Equivalência |
| \oplus | Fusão de ínfons |
| \otimes | Fusão de situações |
| G | Grau de relevância |
| = | Igualdade |
| I | Índice de importância |
| \vdash | Inferência |
| L | Limiar de visibilidade |
| V | Meia vida |
| ∇ | Não é sobre |
| \in | Pertinência |
| P | Potencial de ativação |
| \perp | Preclusão |
| \exists | Quantificador existencial |
| \forall | Quantificador universal |
| | Restrição |
| - | Subtração |
| \models | Suporte |
| Z | Taxa de degradação |
| \cup | União de situações |

Lista de Figuras

| | |
|--|----|
| FIGURA 1.1 - Crescimento do comércio na Internet em bilhões de dólares..... | 14 |
| FIGURA 1.2 - Sistemas de Hiperídia Adaptativa..... | 15 |
| FIGURA 2.1 - Diagrama simplificado de um hiperdocumento..... | 19 |
| FIGURA 2.2 - Posicionamento da HAM em um sistema de hipertexto..... | 21 |
| FIGURA 2.3 - O modelo Trellis de referência para hipertexto (modelo r)..... | 21 |
| FIGURA 2.4 - As camadas do modelo Dexter..... | 22 |
| FIGURA 2.5 - A arquitetura cliente-servidor da web..... | 25 |
| FIGURA 3.1 - O loop clássico: “modelo do usuário - adaptação”..... | 31 |
| FIGURA 3.2 - Espaços de Adaptação em HA Clássica..... | 31 |
| FIGURA 4.1 - Quantização em redes de informação..... | 44 |
| FIGURA 4.2 - Tipos de fechos em grafos de duas setas..... | 46 |
| FIGURA 5.1 - Determinação do grau de relevância entre dois documentos..... | 66 |
| FIGURA 5.2 - Um descritor impreciso..... | 67 |
| FIGURA 5.3 - Um descritor quantizado..... | 68 |
| FIGURA 5.4 - Redes situadas e quantizadas sobrepostas..... | 69 |
| FIGURA 6.1 - Projeções de um objeto nos diferentes planos da HA..... | 72 |
| FIGURA 6.2 - Arquitetura genérica de um sistema de HA proativa..... | 73 |
| FIGURA 6.3 - A interface do usuário..... | 74 |
| FIGURA 6.4 - O processo de modelagem..... | 75 |
| FIGURA 6.5 - O modelo do usuário..... | 76 |
| FIGURA 6.6 - Integração de representações na modelagem do usuário..... | 77 |
| FIGURA 6.7 - O processo de adaptação..... | 77 |
| FIGURA 7.1 - Estrutura do sistema ia@net..... | 82 |
| FIGURA 7.2 - As estruturas estática, funcional e dinâmica em diferentes graus de abstração..... | 84 |
| FIGURA 7.3 - Um descritor para um módulo do curso..... | 88 |
| FIGURA 7.4 - O roteiro de um módulo..... | 90 |
| FIGURA 7.5 - A página de apresentação do sistema..... | 91 |
| FIGURA 7.6 - O Modelo Inicial do Estudante..... | 92 |
| FIGURA 7.7 - O Modelo do estudante após alguma interação com o sistema..... | 93 |
| FIGURA 7.8 - Visão geral do processo de adaptação..... | 95 |
| FIGURA 8.1 - Evolução dos modelos de usuários e do sistema na representação em rede..... | 96 |

Lista de Tabelas

| | |
|---|----|
| TABELA 3.1 - Métodos e Técnicas de Navegação Adaptativa em Sistemas de HA (1) | 37 |
| TABELA 3.2 - Métodos e Técnicas de Navegação Adaptativa em Sistemas de HA (2) | 42 |
| TABELA 4.1 - Operadores de fechamento | 48 |
| TABELA 4.2 - Uma possível configuração dos parâmetros de adaptação | 49 |
| TABELA 4.3 - Meia vida de uma rede em função da sua taxa de degradação | 50 |
| TABELA 5.1 - Operações sobre Ínfons e Situações | 61 |
| TABELA 5.2 - Postulados Básicos do Raciocínio sobre Situações | 63 |
| TABELA 5.3 - Postulados de Combinação | 64 |
| TABELA 5.4 - Postulados Baseados em Ínfons | 65 |
| TABELA 5.5 - Cálculo do grau de relevância entre documentos | 68 |
| TABELA 7.1 - Configuração de hardware utilizada no projeto | 83 |
| TABELA 7.2 - Classificação dos conteúdos do ia@net | 86 |

Resumo

Os Sistemas de Hipermedia (SH) vem se tornando cada vez mais populares em diversas áreas de aplicação, tais como educação, marketing, comércio eletrônico, informação pessoal e serviços inteligentes de interface. Atualmente um dos principais ramos da pesquisa em SH são os Sistemas de Hipermedia Adaptativa (SHA) [BRU 96] [ESP 97], juntamente com as áreas relacionadas de Modelagem do Usuário (MU) e Interfaces Inteligentes (II). Uma das características mais críticas em um SHA é o *modelo do usuário*, uma representação dos objetivos, conhecimento, preferências, necessidades e desejos de seus usuários. A idéia é que usuários com diferentes perfis ou modelos estarão interessados em diferentes peças de informação dentre as apresentadas em uma página hipermedia e podem também desejar navegar no sistema através de diferentes links. A ação adaptativa em um SHA é orientada de modo a oferecer a seus usuários informação hipermedia e navegação ajustados aos respectivos modelos.

A adaptação é geralmente considerada de uma forma *retroativa*, onde as estruturas de apresentação e navegação são produzidas como simples reações à evolução passada do modelo do usuário e à oportunidades oferecidas pelo ambiente. A adaptação *proativa* [PAL 98] adota a idéia de seleção e mesmo a geração *ativa* de hiperdocumentos que serão provavelmente interessante para um determinado usuário. O uso de modelos proativos para a obtenção de informação personalizada permite a antecipação das necessidades e demandas do usuário. Isto é obtido através do emprego de algum tipo de inferência sobre os objetos hipermedia disponíveis, restrita pelo conhecimento disponível no modelo do usuário.

No presente trabalho propõe-se uma metodologia para a construção de SHA, através da integração de dois modelos proativos diferentes. O primeiro desses modelos possui características conexionistas e é orientado à navegação adaptativa. Este modelo destaca a representação comportamental dos links na rede, considerando a frequência com que estes são percorridos. O processo de modelagem é balizado pelas leis de transitividade e reflexividade, que permitem representar proativamente o hiperespaço, simplesmente através da quantização de seus links, abstraindo o conteúdo de seus nodos.

O segundo modelo trata dos aspectos semânticos do processamento de informações através da lógica das situações, que oferece um arcabouço formal para a representação, composição e inferência da relação de relevância entre os nodos de sistemas de hipermedia adaptativa. O ponto de partida são os conceitos de ínfon, documento e descritor, assim como a semântica da relação "é sobre" que pode existir entre diferentes documentos. A integração entre os dois modelos é realizada através da sobreposição das representações em um domínio compartilhado.

Uma arquitetura genérica orientada a agentes para o desenvolvimento de sistemas de HA proativa é apresentada, centrada nos processos de interfaceamento, modelagem e adaptação. A tese se completa com o projeto e desenvolvimento de um sistema educacional online com adaptação proativa para a World Wide Web. Trabalhos futuros são propostos nas áreas da educação, sistemas de informações pessoais e o trabalho colaborativo de equipes.

PALAVRAS-CHAVE: Hipermedia Adaptativa, Modelos Proativos, World Wide Web, Teoria do Fecho, Lógica das Situações, Sistemas Educacionais Adaptativos.

TITLE: "PROACTIVE MODELS FOR ADAPTIVE HYPERMEDIA"

Abstract

Hypermedia Systems (HS) are becoming more and more popular in several application areas, like education, marketing, e-commerce, personal information and intelligent interface services. Currently, one of the main branches in HS research is Adaptive Hypermedia Systems (AHS) [BRU 96] [ESP 97], with its related technologies, like User Modeling (UM) and Intelligent Interfaces (II). One of the most critical features in an AHS is the *user model*, a representation of the goals, knowledge, preferences, needs and desires of its users. The underlying idea is that users with different profiles or models will be interested in different pieces of information presented on a hypermedia page and may also want to use different links for navigation. The adaptation task performed by an AHS is oriented to assist users with personally-tailored hypermedia information and navigation.

Adaptation is in general viewed in a *retroactive way* where presentation and navigational structures are produced as simple reactions to the past evolution of the user model and environment opportunities. The *proactive way* [PAL 98] supports the idea of active selection or even the generation of hyperdocuments that will *probably be interesting* to a particular user. Use of proactive models in personal information gathering allows the anticipation of users needs and requests. This is achieved by applying some kind of inference over the available hypermedia objects, constrained by the knowledge present in the user model.

In this work a methodology for AHS construction of is proposed, by means of the integration of two different proactive models. The first one has a connexionist trait and is oriented to adaptive navigation. This model enhances behavioral representation of the links in the network through the frequency in which they are activated. Modeling process is here controlled by the laws of transitivity and symmetry, allowing proactive representation of the hyperspace only by means of the links, with no regard for the nodes contents.

The second model is related with semantic aspects of information processing through the theory of situations, which offers a formal framework for representing, composing and inference of the *relevance* relationship between nodes in AHS. The starting point here are the concepts of *infor*, document and descriptor, as the semantic of the *aboutness* relationship that may occur between documents. The integration of these two models is done by superposing the representations on a shared domain.

An agent-oriented general architecture for the development of proactive AHS is presented, focusing interface, modeling and adaptation processes. The work is concluded with the project and development of a educational online system with proactive adaptation for the World Wide Web. Future work is proposed in the areas of education, personal information systems and collaborative development.

KEYWORDS: Adaptive Hypermedia, Proactive Models, World Wide Web, Closure Theory, Situation Theory, Educational Adaptive Systems

Capítulo 1

Introdução

1.1 Números

Um dos grandes triunfos do engenho humano, ocorrido nos últimos anos do século XX, foi a consolidação da Internet como um espaço virtual compartilhado, de arquitetura distribuída e alcance global, onde uma crescente variedade de informações, recursos e serviços se encontram disponíveis. A NUA Surveys [NUA 99], empresa européia líder no setor de pesquisas de mercado, estima em 201 milhões o número de pessoas com acesso à Internet em setembro de 1999. Esta cifra, ainda que considerável, está muito aquém do que era previsto em [ZAK 96], onde a estimativa para o ano 2000 era de 500 milhões de usuários online. A NUA Surveys, que se baseia em diversas outras fontes e indicadores para compor um quadro global, prevê ainda que no final do ano 2000 a Internet terá 250 milhões de usuários, o que corresponde a pouco mais de 4% da população da Terra. O mesmo relatório projeta 350 milhões de usuários somente para 2005, o que corresponde a uma notável contração nas explosivas taxas de crescimento da Internet, que até 1996 vinha dobrando o número de usuários a cada ano.

Ao mesmo tempo em que seu crescimento vegetativo tende a atingir um patamar de estabilidade ao redor dos 15 ou 20 por cento [NUA 99], outros aspectos da Internet passam a apresentar desenvolvimento acelerado. Um exemplo típico é o comércio eletrônico ou *e-commerce*, cujo crescimento estimado em bilhões de dólares até 2002 é apresentado na Figura 1.1.

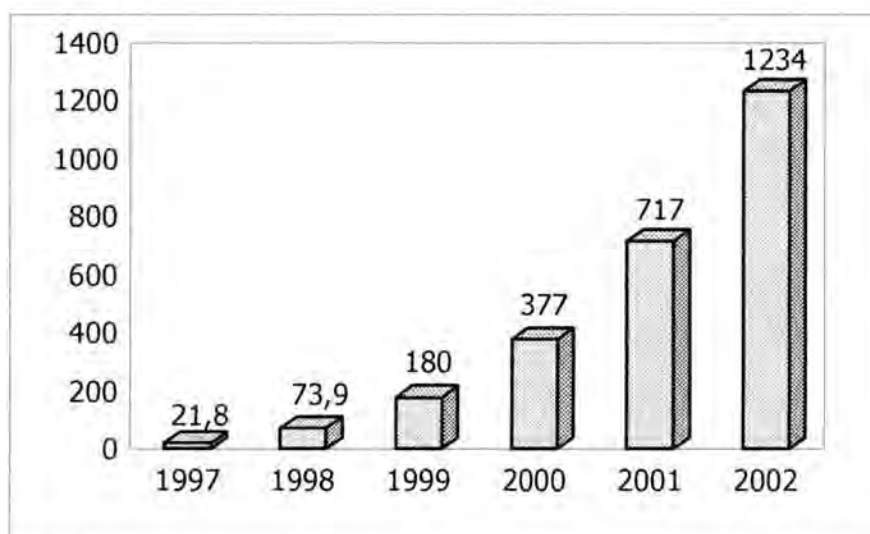


FIGURA 1.1: Crescimento do comércio na Internet em bilhões de dólares. Fonte: [NUA 99]

1.2 Hipermissão Adaptativa

Na Internet o uso de hipertexto e hipermissão na comunicaço de ideas, informaço e prestaço de serviços tornou-se consagrado a partir da expanso da World Wide Web (WWW ou simplesmente web), a teia global cuja estrutura conceitual e de navegaço e baseada na noço de hipertexto. As aplicaçoes desenvolvidas para a web atingem um numero muito maior de usuarios do que as aplicaçoes interativas tradicionais. Uma alternativa possivel a abordagem da *interface unica*, identica para todos os usuarios [STA 97], seria a adaptaço dos conteudos das paginas - e dos objetos hipermissão que elas permitem acessar - as necessidades dos usuarios individuais.

A area da *hipermissão adaptativa* (HA) tem despertado grande interesse nos ultimos anos, fazendo crescer no cenario internacional o numero de workshops e congressos onde o tema e tratado com destaque. Em paralelo, diversas novas areas de aplicaçoes de sistemas de HA tem sido identificadas, incluindo entre outras, alem do comercio eletronico, marketing, educaço, sistemas de informaçoes, medicina, publicidade e lazer. Os sistemas *adaptativos* mantem um modelo com diversas caractersticas dos seus usuarios, tais como objetivos, interesses, preferncias, nivel de conhecimento, etc. Este modelo, denominado o *modelo do usuario*, e empregado como referncia para a construço de paginas especialmente adaptadas a cada usuario do sistema. Diversos metodos e tecnicas tem sido empregados com este objetivo, seguindo duas principais linhas de adaptaço: a *apresentaço adaptativa* e a *navegaço adaptativa* [BRU 96] [BRU 97].

O principal objetivo das pesquisas em HA e o estudo e elaboraço de representaçoes adequadamente expressivas, economicas e eficientes, capazes de suportar nao so os aspectos estaticos e funcionais do processo de adaptaço, mas tambem suas caractersticas dinamicas, organizacionais e comportamentais. Tudo isso torna a modelagem do processo de adaptaço em sistemas de HA uma tarefa complexa e dificil, de cuja preciso entretanto dependem os resultados que se deseja atingir. Para empregar determinada tecnica os sistemas adaptativos buscam identificar padroes no modelo do usuario que justifiquem a execuço de determinada *funço de adaptaço*. Em um nivel muito alto o relacionamento entre os principais processos em sistemas de HA pode ser visto na figura 1.2.

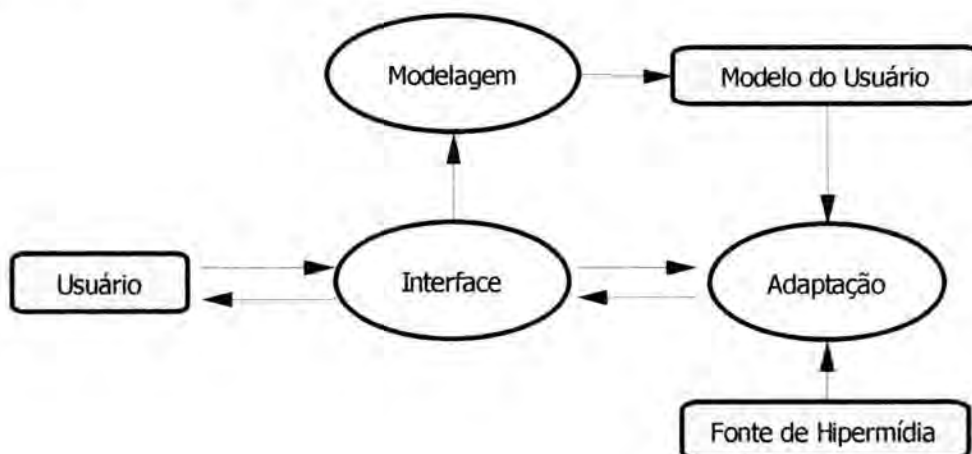


FIGURA 1.2: Sistemas de Hipermissão Adaptativa

1.3 Modelos de Adaptação na Web

A web adota o princípio da distribuição na representação de conhecimento, o que significa que este é armazenado como uma rede de nodos e links. Os nodos podem conter qualquer combinação de texto, imagens, som, vídeo, etc., com os links formando algo assim como uma rede de conceitos conectados através de relacionamentos informais. Os usuários *navegam* nesta rede seguindo os links que lhes são significativos entre um nodo e outro. Neste processo empregam uma forma de julgamento associativo que visa conduzir a partir de uma certa posição inicial ao nodo ou nodos que contém a informação desejada.

Nesta visão a organização *material* da web é produzida por seus projetistas, que usam suas idéias intuitivas de estruturação do conhecimento e semântica na construção de nodos e sub-redes. Estas contribuições individuais são integradas gradualmente a corpos mais amplos de conhecimento preexistentes na web, assim expandindo o conhecimento da rede. Em contraposição à organização material, o que se poderia denominar organização *virtual* da rede é produzida pelo processo dinâmico de navegação, o qual decorre da percepção associativa dos usuários. No atual estágio tecnológico da web o conhecimento produzido pelo processo de navegação não é registrado sistematicamente, no entanto poderia ser empregado na construção de modelos de aprendizado associativo, onde a web é a metáfora de uma memória associativa de grandes dimensões [BOL 96].

Já sob uma ótica simbólica, o processo de navegação na web corresponde a trilhar um caminho na teia de conexões escolhendo a cada passo um particular link dentre todas as possibilidades oferecidas na página. O usuário pode sempre retornar (*back*), interromper a navegação (*home* ou *quit*), saltar para um local preferido (*bookmarks*), etc., mas o princípio da ação é sempre a *busca por alguma coisa que é relevante aos objetivos do usuário*. O foco desta busca encontra-se exatamente na escolha da próxima conexão, que deverá levar o usuário para mais próximo da satisfação de seus objetivos. Ao fazer esta escolha, o usuário aplica intuitivamente noções que lhe permitem decidir *que determinado link é mais relevante que os demais para os seus objetivos e portanto deve ser selecionado para continuar a navegação*.

A noção de que uma determinada peça de informação é “*sobre*” alguma coisa está intimamente associada com a *relevância* que a informação apresenta em determinada *situação*. Se uma peça de informação é *sobre* uma necessidade manifesta de informação (usualmente uma *consulta*) então esta peça é considerada *relevante* com respeito à informação solicitada. São importantes aqui os conceitos relacionados com a qualificação da *relevância situada* de uma determinada peça de informação a partir da teoria geral da informação [HUI 94]. Esses conceitos surgem da *Teoria da Situação* [DEV 91] que abrange uma teoria da informação suficientemente poderosa para estudar a relação entre certa informação e aquilo *sobre* o que ela é.

Os estudos realizados sobre sistemas de HA na web (ver capítulo 3) levaram a observação de que os modelos utilizados são em geral *retroativos*, isto é, a adaptação ocorre sobre informações de eventos passados. Uma alternativa para essa estratégia seria o emprego de modelos de adaptação *proativos*, capazes de *antecipar* os objetivos e necessidades dos usuários. O trabalho realizado na presente tese contempla o estudo e desenvolvimento de modelos de adaptação proativa para sistemas de HA na web. Duas alternativas foram consideradas, uma *conexcionista*, baseada na aplicação dos fechos

simétrico e transitivo sobre redes quantizadas, e outra *simbólica*, baseada na lógica das situações e na relação de relevância entre os conteúdos das páginas da rede considerada.

As duas visões - associativa e simbólica - são claramente complementares na elaboração de modelos de usuários, tendo em vista sua capacidade de representar aspectos diferentes do mesmo. As duas abordagens atuando sobre diferentes níveis de uma mesma representação estabelecem uma relação naturalmente cooperativa e mutuamente catalisadora, onde os resultados obtidos por qualquer uma delas podem auxiliar na produção de resultados na outra.

1.4 Objetivos da Pesquisa

A pesquisa desenvolvida na presente tese contempla quatro objetivos bem definidos relacionados abaixo:

1. Revisão bibliográfica sobre modelos de hipertexto, a web e hipermedia adaptativa;
2. Estudo, desenvolvimento e integração de um modelo simbólico e de um modelo conexionista para a representação de sistemas adaptativos na web;
3. Proposta e desenvolvimento de uma arquitetura orientada a agentes para sistemas adaptativos na web;
4. Implementação de um sistema educacional adaptativo genérico baseado no modelo e arquiteturas propostos.

1.5 Organização da Tese

Esta tese está organizada da seguinte maneira: os capítulos 2 e 3 são de revisão bibliográfica e contém o relato dos estudos realizados sobre os modelos clássicos de hipertexto, a World Wide Web e hipermedia adaptativa, incluindo métodos e técnicas de navegação e apresentação adaptativas. No capítulo 4 desenvolve-se um modelo associativo, baseado em redes quantizadas e na teoria do fecho para representar os processos dinâmicos da navegação na web. O relacionamento de relevância entre as páginas é introduzido no capítulo 5, com base na teoria da situação. A integração entre as duas representações é obtida através da sobreposição dos modelos a partir das entidades que são nodos em ambos. No capítulo 6 apresenta-se a proposta de uma arquitetura genérica orientada a agentes para sistemas adaptativos na web, projetada para suportar as representações simbólica e conexionista na modelagem da adaptação. O sistema educacional genérico *ia@net* é descrito no capítulo 7, onde são discutidos os principais aspectos de sua implementação. As conclusões do trabalho são apresentadas no capítulo 8, onde também são consideradas possíveis propostas de trabalho futuras em continuidade à pesquisa realizada.

Capítulo 2

Modelos de Hipertexto e a World Wide Web

No presente capítulo apresentam-se os principais conceitos e os modelos clássicos empregados em sistemas hipertexto e hipermedia, incluindo a Máquina Abstrata de Hipertexto (HAM), e os modelos Trellis, Dexter, Lange e Torre. Uma revisão técnica da World Wide Web (WWW ou web) é formulada, sendo apresentados de modo sumário sua arquitetura cliente-servidor, sua interpretação como um sistema hipertexto e sua estrutura fundamental de sustentação, incluindo aqui o esquema de endereçamento empregado, a linguagem HTML e o protocolo HTTP. No final do capítulo a web é analisada do ponto de vista dos modelos apresentados.

2.1 Hipertexto e Hipermedia

A última década testemunhou o surgimento e a maturação de diversas tecnologias para representação, armazenamento, processamento e recuperação de informações. Tais tecnologias encontram-se em grande parte apoiadas na idéia de hipertexto, originalmente introduzida no sistema *Memex* proposto por Vannevar Bush em seu artigo clássico "As We May Think" [BUS 45]. O termo *hipertexto* apareceria vinte anos mais tarde, em 1965, cunhado por Ted Nelson para designar a estrutura implementada em *Xanadu*. Foi, entretanto somente com o desenvolvimento dos sistemas *Guide* em 1986, *Hypercard* em 1987 e a realização neste mesmo ano da *Primeira Conferência sobre Hipertexto*, organizada pela ACM, que a pesquisa em hipertexto se estabeleceu como um campo bem definido de investigação.

O uso de hipertexto tornou-se popular na Internet através da World Wide Web (WWW ou simplesmente web), concebida por Tim Berners Lee no CERN em 1989. A noção de hipertexto desempenha para a web um papel fundamental, na medida em que estabelece sua organização e dinâmica básicas, oferecendo uma nova dimensão à representação de fatos, idéias e conceitos. Na web a noção original de hipertexto é ampliada pela incorporação de objetos multimídia estruturados em *hiperdokumentos*, cujo conteúdo, que pode abranger texto, imagens, áudio, vídeo, etc., recebe a denominação genérica de *hipermedia*.

Uma definição interessante de hipertexto pode ser encontrada em [SCH 89], onde um sistema hipertexto é visto como "*uma base de dados possuindo conexões ativas que permitem ao usuário saltar de um ponto para outro em seu interior, conforme desejado*". Esta definição permite entender hipertexto a partir dos seguintes pontos:

- Um hipertexto é uma *base de dados*. A informação contida em um hipertexto não é simplesmente uma massa informe de dados. Ao contrário, possui uma estrutura semelhante à encontrada nas informações armazenadas em bases de dados convencionais.
- A ação típica do usuário é o *salto* entre dois pontos distintos da base de dados. Isto difere do que acontece em bases de dados convencionais, onde a ação usual

é a formulação de consultas que demandam a apresentação de elementos armazenados em diferentes pontos da base de dados.

- Um sistema hipertexto contém, além de dados, uma estrutura de *conexões* representando relações não especificadas entre as peças de informação que armazena. Essas conexões funcionam como um guia para a navegação do usuário entre diferentes pontos da base de dados.

Ainda que um sistema de bases de dados possa ser empregado para o armazenamento e recuperação das informações contidas em um hipertexto, sua interface relacional usual deve ser substituída por uma nova, capaz de suportar a navegação entre os conteúdos do hipertexto. Interfaces com esta capacidade são tipicamente denominadas *browsers*. As peças de informação ou partes da base de dados são denominadas *nodos* enquanto as conexões entre nodos chamam-se *links*. Juntos, nodos e links constituem o que se denomina um *hiperdocumento*. A estrutura de nodos e links pode ser vista como um grafo, o qual pode assumir uma complexidade arbitrária. Na Figura 2.1 apresenta-se uma visão simplificada de um pequeno hiperdocumento com cinco nodos e sete links. Na mesma figura pode-se ver também que os links apontam para pontos específicos - palavras ou regiões especiais no interior de um nodo - que recebem a denominação de *âncoras*.

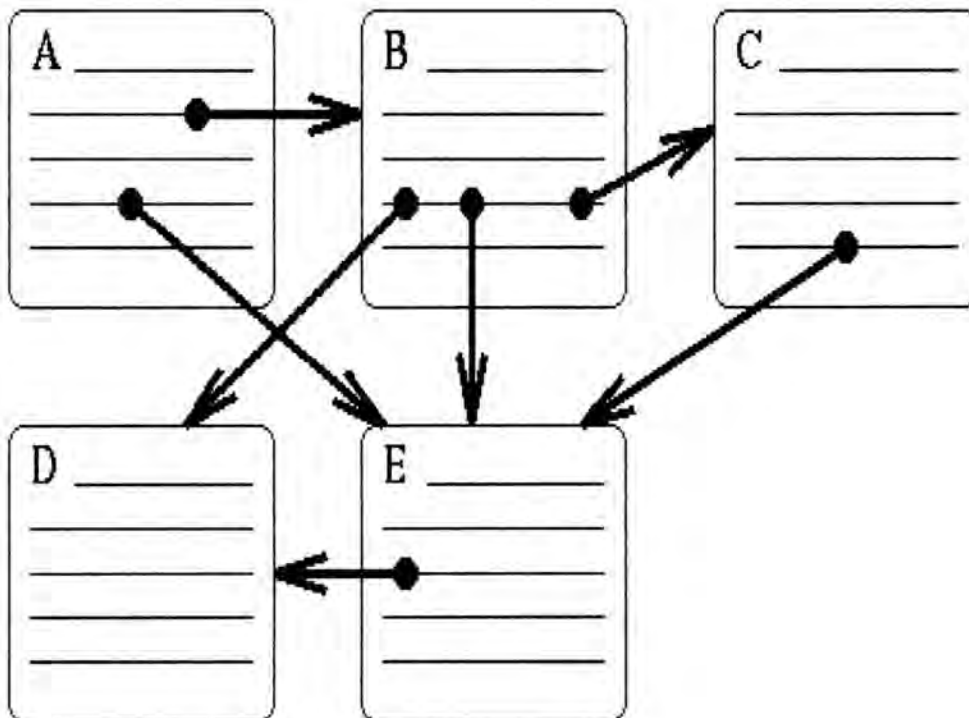


FIGURA 2.1 - Diagrama simplificado de um hiperdocumento

O termo hipermidia é um acrônimo que combina as expressões hipertexto e multimídia. Segundo De Bra, hipermidia e hipertexto são freqüentemente encontrados na literatura como sinônimos [DeB 97]. Ainda que a palavra hipertexto possa sugerir que toda a informação disponível encontra-se sob a forma de texto, a maioria dos sistemas de hipertexto permite o uso de outras formas de representação, incluindo gráficos, sons, animação, vídeo, etc. No presente trabalho optou-se por empregar as

duas expressões como sinônimos, no sentido de que designam o mesmo objeto ou sistema. Preferiu-se, entretanto adotar informalmente o uso de *hipertexto* quando o foco do discurso recair sobre os aspectos sintáticos, estruturais e operacionais do sistema e *hipermídia* quando se tratar dos aspectos semânticos e organizacionais do seu conteúdo. Assim cada uma das expressões estará associada a uma particular manifestação do mesmo objeto de estudo.

2.2 Modelos para Sistemas de Hipertexto

Um *sistema de hipertexto* é uma peça de software de considerável complexidade, constituída por diversos componentes que atuam concorrentemente na execução de diferentes funções. Ao longo dos anos diversos modelos foram propostos para representar sua arquitetura. Tais modelos são denominados *modelos de referência*, uma vez que oferecem uma estrutura conceitual abstrata para a construção e operação de sistemas de hipertexto. Os principais modelos de referência para sistemas de hipertexto são os seguintes [DeB 97]:

- *HAM* ou Hypertext Abstract Machine, [CAM 88];
- O modelo Trellis, proposto por Stotts e Furuta [STO 89];
- O modelo Dexter, especificado em Z por Halasz e Schwartz [HAL 90];
- Um modelo formal de hipertexto, em VDM por Danny B. Lange [LAN 90];
- O modelo Tower, orientado a objetos, por De Bra, Houben e Komatzky [DeB 92].

Esses modelos são brevemente comentados a seguir, tendo em vista o interesse nas diferentes soluções que propõem para a modelagem de sistemas de hipertexto.

2.2.1 HAM ou Hypertext Abstract Machine

A HAM é um servidor baseado em transações para um sistema de armazenamento hipertexto. O servidor é projetado para suportar múltiplos usuários em um ambiente em rede. O sistema de armazenamento indexa uma coleção de contextos, nodos, links e atributos que configuram o grafo do hipertexto. A definição da HAM consiste na descrição de *objetos HAM* e de um conjunto de *operações* que sobre eles podem ser aplicadas [CAM 88]. Como acontece com a maioria dos modelos de referência, a HAM não descreve completamente um hipertexto, situando-se entre o sistema de arquivos e a interface do usuário. A Figura 2.2 mostra a representação gráfica do seu posicionamento em um sistema hipertexto.

Na mesma figura, vemos que a HAM é um modelo de representação de baixo nível, fortemente vinculada ao sistema de armazenamento da máquina hospedeira e com um relacionamento mais frouxo com a aplicação e a interface do usuário. Fica claro também que a HAM é apenas parte do sistema de hipertexto, que por sua vez é formado pelos quatro componentes apresentados na figura.



FIGURA 2.2 - Posicionamento da HAM em um sistema de hipertexto [CAM 88].

2.2.2 O Modelo Trellis

Richard Furuta e David Stotts desenvolveram um sistema de hipertexto baseado em Redes de Petri que foi denominado *Sistema Trellis*. Este implementava o modelo Trellis de hipertexto, proposto em [STO 89]. A partir do modelo Trellis os autores deduziram um meta-modelo - o *Modelo Trellis de Referência para Hipertexto*, ou abreviadamente *modelo r* [STO 90].

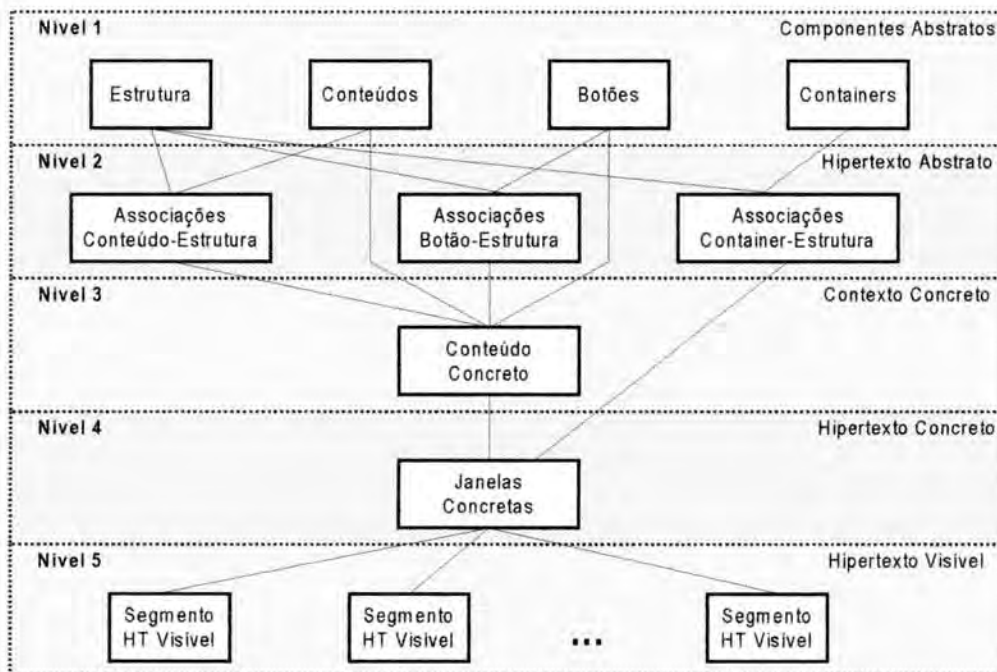


FIGURA 2.3 - O modelo Trellis de referência para hipertexto (modelo r)

O modelo *r* é composto por cinco camadas lógicas, como é mostrado na Figura 2.3. Em cada nível encontram-se uma ou mais representações de parte ou de todo o hipertexto. Ao contrário da HAM e de outros modelos de referência, as camadas representam *níveis de abstração* e não componentes do sistema. Os níveis, por sua vez, são agrupados em três categorias: abstratos, concretos e visíveis, em correspondência com o modelo de Shaw para documentos impressos [SHA 80].

Na Figura 2.3 os arcos indicam que um conceito em um nível mais baixo é dependente de um conceito em um nível mais alto de abstração. Para a aplicação do modelo *r* em um sistema de hipertexto real é necessário que os componentes deste último sejam representados através dos níveis de abstração do modelo *r*. Isto pode ser difícil, uma vez que certos componentes podem somente ser descritos em determinado níveis de abstração. Stotts e Furuta demonstraram entretanto a modelagem de estruturas, âncoras, diversas classes de links e conteúdos dinâmicos.

2.2.3 O Modelo Dexter

O modelo Dexter [HAL 90] é uma tentativa de capturar formal e informalmente um importante conjunto de abstrações encontradas em diferentes sistemas de hipertexto e suas implementações. O objetivo do modelo foi o de oferecer uma base inicial para a comparação de diferentes sistemas de hipertexto assim como desenvolver padrões de intercâmbio e interoperabilidade. O modelo divide-se em três camadas, conforme é mostrado na Figura 2.4.



FIGURA 2.4 - As camadas do modelo Dexter

O principal foco do modelo Dexter encontra-se na camada de armazenamento, que modela o hipertexto como uma rede composta de nodos e links. Esta camada descreve uma base de dados composta por uma hierarquia de componentes - os nodos - que se interconectam através de links relacionais. Aqui são encontrados os mecanismos que permitem aos nodos e links integrar-se na rede do hipertexto. Os componentes são tratados nesta camada como containers genéricos de dados. Já a camada dos componentes internos está relacionada com os conteúdos e estruturas encapsuladas nos nodos. Esta camada é propositadamente excluída do modelo Dexter e seus componentes são apenas indexados aos nodos na camada de armazenamento por meio de *âncoras*.

As camadas de armazenamento e dos componentes internos tratam hipertextos como estruturas de dados essencialmente passivas. O sistema de hipertexto entretanto oferece mecanismos para acessar, visualizar e manipular a rede subjacente. Esta funcionalidade é capturada pela camada de runtime, da qual o modelo Dexter permite descrever somente a estrutura básica das estruturas de apresentação. O interface entre a camada de runtime e a camada de armazenamento é obtido através da noção de *especificações de apresentação* que refletem a forma com que os componentes na camada de armazenamento devem ser apresentados ao usuário na camada de runtime.

2.2.4 Um Modelo Formal de Hipertexto

Este modelo foi concebido por Danny B. Lange, que o apresentou no Primeiro Workshop sobre Padronização de Hipertexto. Foi introduzido em [LAN 90]. O modelo é formal, uma vez que foi descrito na linguagem de especificação VDM (Vienna Development Method). Segundo o autor, a principal motivação para o desenvolvimento era solucionar a dificuldade de intercâmbio e comunicação entre os sistemas de hipertexto existentes. A linguagem VDM foi escolhida por permitir o desenvolvimento de especificações de software de forma top-down, facilitando assim sua verificação formal.

Assim como o modelo Dexter, o formalismo proposto por Lange enfatiza a estrutura de dados do hipertexto, sendo denominado pelo autor o *modelo de dados hipertexto*, onde são definidos nodos, links, estruturas de rede, etc. O modelo formal vai além do modelo Dexter uma vez que trata aspectos dos conteúdos dos nodos, introduzindo aos conceitos de slot, botão e campo. O modelo de dados é então estendido com as abstrações da orientação a objetos para hipertexto.

Em decorrência da concentração do modelo formal na modelagem dos dados, as camadas de apresentação e a semântica de navegação não são por ele tratadas. As aplicações somente operam sobre o hipertexto através de operações previamente definidas e os objetos de dados não têm consciência das aplicações e sua semântica. Por outro lado, acrescentando-se o aspecto de persistência ao modelo orientado a objetos, obtém-se a especificação formal de uma base de dados hipertextual orientada a objetos. Seguindo esta linha, questões de distribuição, versão e controle de acesso seriam tratadas no domínio dos sistemas de gerenciamento de objetos. Uma séria restrição do modelo de Lange é sua forte orientação à informação textual. Isto por um lado torna possível uma descrição mais precisa e detalhada da estrutura interna dos nodos, entretanto torna o modelo inadequado para o desenvolvimento de sistemas hipermídia em geral. Além disso algumas descrições detalhadas presentes no modelo reduzem sua generalidade mesmo para hipertextos totalmente textuais.

2.2.5 O Modelo Torre

Os autores dos modelos Trellis e Dexter deixaram diversas questões em aberto devido à sua dificuldade em prever a evolução futura dos mecanismos presentes em hipertextos. O modelo Trellis, por exemplo, descreve a camada dos componentes abstratos de uma forma que o mantém coerente mesmo quando não há necessidade de objetos contendo outros objetos. O modelo Dexter por sua vez prevê links indiretos e bidirecionais, mas somente entre dois nodos (denominados componentes). Links entre mais de dois nodos são permitidos mas devem ser obrigatoriamente direcionados. Outra restrição do modelo Dexter é que, apesar de permitir que objetos compostos contenham outros objetos compostos, a hierarquia resultante deve ser acíclica, impedindo o surgimento do fenômeno conhecido como *efeito Escher*.

Na quarta Conferência da ACM sobre Hipertexto, De Bra, Houben e Komatzky propuseram um novo modelo de dados para hipertexto, orientado a objetos e extensível [DeB 92]. O modelo definia as estruturas básicas correspondentes a nodos, links e âncoras, além de objetos-torres e objetos-cidades. Um objeto-torre captura diferentes descrições de um mesmo objeto, de modo semelhante às camadas no modelo Dexter. Tipos de dados, estruturas de armazenamento, apresentações etc são níveis do modelo Torre. Objetos-cidades, por sua vez são, conjuntos de visões sobre objetos-torres. É permitido a qualquer objeto ser do tipo *virtual*, isto é, obtido como resultado de uma função ou algoritmo. Também são definidos os operadores *aplicar-a-todos*, *filtrar*, *enumerar* e *abstrair* ou *agrupar*. A semântica de navegação é dada por meio de *trajetórias*, uma generalização do conceito de *caminho*, definido em termos de axiomas espaço-temporais.

2.3 Revisão Técnica da World Wide Web

A World Wide Web - a teia ao redor do mundo, WWW ou simplesmente web - é uma das mais importantes inovações tecnológicas do século XX. Inventada em 1989, a web é o mais importante veículo de acesso, compartilhamento e transferência de informações e recursos sobre a Internet. Na web é possível encontrar informação sobre quase tudo e também disponibilizar instantaneamente qualquer informação ou recurso digital aos seus milhões de usuários. Estes têm na web um sistema hipertexto armazenando um imenso repositório de documentos densamente conectados através de *links* que podem ser percorrido ou navegado a partir de qualquer ponto. Isto é obtido por meio de um *browser*, um programa executado localmente que permite ao usuário seguir os links de um documento para outro sem que seja necessário nenhum conhecimento técnico ou especial.

Ao mesmo tempo em que cresce rapidamente em número de usuários, hosts e volume de documentos, a web também evolui tecnologicamente em marcha acelerada. Uma grande quantidade de recursos tem sido investida em pesquisa e desenvolvimento de tecnologias para a web. Segundo Loke, as principais áreas pesquisadas incluem protocolos de comunicação, formatação de dados, software, segurança e privacidade. [LOK 98] Nos últimos anos tem-se notado um crescente interesse em adicionar conteúdo computacional à web, com vistas à apresentação de documentos mais interessantes, expressivos e interativos, permitindo através deles a realização de tarefas mais complexas e produtivas.

As principais tecnologias que dão suporte à web podem entretanto ser aglutinadas em quatro focos principais:

1. Arquiteturas cliente-servidor;
2. Esquemas de endereçamento;
3. Linguagens para construção de hiperdocumentos;
4. Protocolos de rede.

Na presente seção estas tecnologias são examinadas em maior detalhe, visando desenvolver uma visão mais precisa da web e de seus mecanismos fundamentais.

2.3.1 Arquitetura Cliente-Servidor

A web, como a própria Internet, baseia-se na arquitetura cliente-servidor. Um servidor é um programa que aceita requisições de diferentes usuários através da rede, executa as tarefas requisitadas em seu próprio ambiente e remete as respostas aos usuários que as requisitaram. Um cliente é um programa que pode enviar uma requisição a um servidor, aguardar uma resposta e sinalizar seu recebimento ao servidor. O cliente também aceita entradas e apresenta os dados obtidos ao usuário, atuando como intermediário entre este e o servidor. O acesso à informação na Internet é assim controlado por servidores, que a recuperam a partir das requisições recebidas dos clientes.

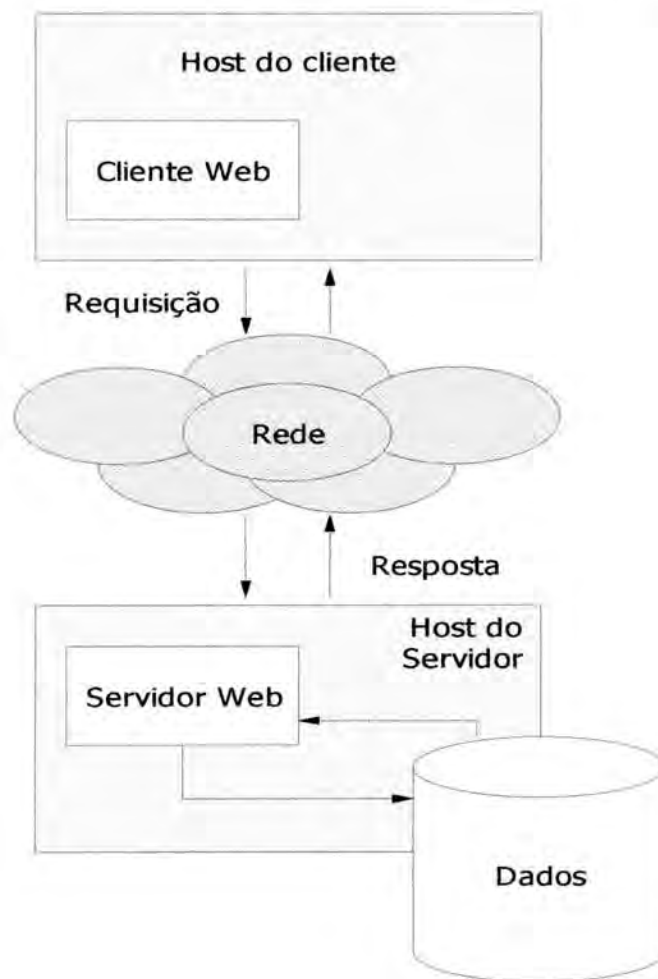


FIGURA 2.5 - A arquitetura cliente-servidor da web

Servidores e clientes podem ser caracterizados pelo protocolo que utilizam. No caso da web o protocolo adotado é o hypertext transfer protocol (http). Os clientes web - ou browsers - mais populares como o Netscape, o NCSA Mosaic e o Internet Explorer podem também se comunicar com servidores que implementam outros protocolos como ftp, nntp, wais, etc. O termo *host* denomina a máquina hospedeira onde o cliente e o servidor são executados. Deve-se notar que um cliente e um servidor podem ser executados no mesmo host, isto é, o host pode abrigar simultaneamente o programa cliente e o respectivo servidor. A Figura 2.5 apresenta a arquitetura cliente-servidor básica da web.

2.3.2 A Web como um Sistema Hipertexto

O crescimento acelerado da web deve-se em grande parte ao seu uso como um sistema hipertexto, onde duas coleções de informações independentemente desenvolvidas podem ser facilmente conectadas se algum relacionamento relevante for encontrado entre elas. Da mesma forma, informação recentemente gerada pode ser incorporada a coleções previamente existentes. Este processo de conexão de peças de informação é simplificado pela adoção de um esquema de endereçamento e um formato de documentação padronizados. De acordo com [BER 94] a web pode ser vista como o corpo de dados disponibilizado (materializado) na Internet por meio dos três seguintes itens:

1. Um sistema de endereçamento (denominado *Uniform Resource Identifier - URI*) permitindo a associação de uma referência padronizada a cada diferente item da web;
2. Uma linguagem de representação (denominada *Hypertext Mark-up Language - HTML*), na qual os documentos são representados.
3. Um protocolo de rede (denominado *Hypertext Transfer Protocol - HTTP*) para a recuperação de informações na web;

Estes três itens implementam funcionalmente a web como um sistema hipertexto. Dada a sua importância para o desenvolvimento e contextualização do presente trabalho, discorre-se a seguir brevemente sobre cada um deles.

Uniform Resource Locator (URL)

Uma URL (Localizadora Uniforme de Recurso - tradução forçada para suportar o gênero feminino consagrado em português: "uma URL..."), é um string cujo conteúdo referencia de forma inequívoca um recurso na Internet, contendo instruções explícitas sobre como este deve ser acessado, incluindo o protocolo que deve ser usado e a localização física do recurso. De um modo geral uma URL consiste nos seguintes componentes:

1. O protocolo da Internet com o qual o recurso deve ser acessado;
2. O nome do host do servidor onde o recurso está armazenado;
3. (Opcionalmente) o número da porta através da qual o servidor recebe as requisições no protocolo especificado;
4. O caminho identificando a localização do recurso na árvore de diretórios do servidor;

5. (Opcionalmente) o identificador de uma âncora no interior do hiperdocumento que suporta o recurso.

As URLs contém as URIs (Uniform Resource Identifiers), que são o conjunto de todos os nomes ou endereços que referenciam recursos. Além delas há um outro tipo de URI, ainda em desenvolvimento, que funcionam como um nome persistente para um recurso, sendo denominadas URNs (Uniform Resource Names). Os URNs foram propostos como um esquema de denominação visando evitar os problemas encontrados com a associação de um recurso a um endereço físico, produzida pelas URLs. O uso de URLs torna difícil mudar ou mesmo replicar a localização física de um recurso, uma vez que isso torna necessário a atualização de todos os documentos a ele conectados. A proposta das URNs é identificar um recurso independentemente de sua localização física, tornando a localização física transparente para fins de endereçamento. Dado o seu estágio de desenvolvimento os URNs não são ainda empregados universalmente.

Hypertext Mark-up Language (HTML)

HTML é a linguagem padrão com a qual os documentos da web são escritos. Sua sintaxe baseia-se no uso de *tags* que demarcam e permitem associar atributos diversos aos diferentes componentes do documento. As tags permitem a estruturação do conteúdo do documento em parágrafos, níveis de cabeçalhos, listas ordenadas e não-ordenadas, tabelas, etc. Além disso permitem estabelecer no documento um interface com o usuário, através de menus, campos para entrada de texto, botões, etc. As tags também permitem o encapsulamento de código móvel (p.ex: Java applets ou scripts CGI) que é executado do lado cliente como parte integrante do documento que o transporta. O termo *página web* ou simplesmente *página* é usado para designar documentos html que podem ser acessados através de uma URL.

Longe de ser um trabalho acabado o HTML continua sendo desenvolvido, introduzindo novas tags a cada versão. A versão 4.0, por exemplo, trouxe diversos aperfeiçoamentos incluindo tabelas, applets, fluxo de texto ao redor de imagens, subscrito e sobrescrito e suporte à programação de scripts [LOK 98].

Hypertext Transfer Protocol (HTTP)

Este protocolo baseia-se no esquema de endereçamento por meio de URLs e apresenta uma elevada taxa de transferência viabilizando a navegação em hiperdocumentos distribuídos na Internet. O HTTP permite a transferência de informação nos mais diversos formatos, incluindo texto, hipertexto, imagens e os diversos tipos de mídia descritas no padrão MIME (Multipurpose Internet Mail Extensions).

O HTTP estabelece que a conexão com a Internet deve ser mantida apenas durante uma operação (p. ex: a requisição e transferência de um arquivo) e apresenta seus resultados com base exclusivamente na última requisição do usuário, sem considerar requisições anteriores. Desta maneira o HTTP não precisa oferecer suporte completo à navegação como o FTP (File Transfer Protocol). Uma vez estabelecida a conexão com o servidor, o cliente envia sua requisição, à qual o servidor responde com a informação requisitada (se esta estiver disponível). As requisições emitidas pelos clientes compõem-se de duas partes. A primeira delas contém:

1. O método de requisição, que especifica a operação a executar;

2. A URI do recurso no qual a operação deve ser executada;
3. A versão do protocolo utilizado (p.ex: HTTP 1.1).

A segunda parte consiste em cabeçalhos (opcionais) com informações ao servidor tais como os tipos de objetos que o cliente pode manipular, a identidade do cliente, a URL do documento que originou a requisição, etc.

Os três métodos de requisição implementados mais freqüentemente são o GET que é utilizado na recuperação de um recurso, HEAD, que requisita cabeçalhos com informação sobre um determinado recurso e POST que permite ao cliente enviar informações ao servidor.

Do ponto de vista do cliente a web possui uma natureza não-determinística, no sentido em que ele geralmente não pode prever o resultado de uma requisição enviada a um servidor HTTP. O desempenho e confiabilidade da web são afetados por diversos fatores, por vezes imprevisíveis e fora do controle do usuário. Tais fatores incluem o tráfego na rede, a largura de banda, a disponibilidade do servidor, mudanças na informação disponibilizada, permissão de acesso removida e assim por diante. Por exemplo, o cliente não recebe um arquivo requisitado porque ele foi removido ou o servidor está ocupado. Entretanto uma requisição que falhou pode ser bem sucedida em outro momento.

Em qualquer momento no tempo uma URL referencia uma única página, porém a mesma URL usada em diferentes momentos pode produzir resultados diferentes. A natureza dinâmica da informação na web torna geralmente impossível para o cliente prever qual informação será retornada por uma requisição bem-sucedida. Um exemplo é uma URL que encerra uma consulta a algum mecanismo de pesquisa. O resultado pode variar dependendo do momento em que a consulta foi realizada. Outro exemplo seria a própria atualização das páginas por seus possuidores.

O HTTP está em uso desde 1990 e se encontra em permanente desenvolvimento. O padrão recomendado mais recente é o HTTP 1.1, que foi projetado para aperfeiçoar diversos aspectos do HTTP 1.0, incluindo melhoria no desempenho dos clientes e servidores web e a redução no tráfego devido a transferência de dados com HTTP. Para que estes aperfeiçoamentos tenham efeito, é necessário entretanto que clientes e servidores implementem o HTTP 1.1, o que ainda se encontra em andamento.

2.4 A WWW e os Modelos Clássicos de Hipertexto

Dos três componentes da web é através da linguagem HTML que a sua dimensão hipertexto é mais fortemente representada. A web entretanto possui características únicas de distribuição e abrangência que não são cobertas por todos os modelos clássicos de hipertexto. A HAM, por exemplo, atuando entre o sistema de armazenamento e a camada da aplicação não trata os aspectos de interface ou do sistema de endereçamento distribuído típico da web.

Já o modelo Trellis, ao introduzir os níveis *abstrato*, *concreto* e *visível* na representação de hipertexto, componentes e contextos oferece um ponto de vista que permite inclusive tratar aspectos dinâmicos do sistema modelado. O problema com este modelo é que os componentes do sistema somente podem ser representados através de suas camadas de abstração, o que pode interferir com certas necessidades de modelagem que exigem uma visão integral dos componentes. O modelo Dexter, pela sua

generalidade, oferece uma possibilidade mais ampla e uma profundidade maior de modelagem, permitindo representar desde a camada dos componentes internos até o interface com o usuário. O foco de modelo Dexter reside, entretanto, na camada de armazenamento e, como os demais modelos clássicos, ele também não consegue capturar com clareza o conjunto de ações assíncronas e distribuídas que caracterizam a dinâmica da web.

Em comum todos esses modelos desenvolvem uma noção muito restrita de certos conceitos como *link*. Este componente da web - e de todo sistema hipertexto - é em geral representado como mero conector entre os nodos, que recebem toda a carga semântica do modelo. Um objetivo de pesquisa é portanto produzir novos modelos com links capazes de expressar relacionamentos complexos, além da idéia de simples ligação entre dois nodos.

Outro fator a ser considerado é o surgimento de novos conceitos como *intranet* e *extranet* que introduziram uma nova divisão no hiperespaço. A rede perde assim sua característica homogênea para tornar-se diferenciada, orgânica. Os modelos clássicos, concebidos para representar uma visão restrita e predominantemente estática de hipertexto foram ultrapassados em pouco tempo pela própria realidade, que tem na web possivelmente o seu maior desafio.

Capítulo 3

Hipermídia Adaptativa

Hipermídia Adaptativa (HA) é a área da ciência da computação que se ocupa do estudo e desenvolvimento de sistemas, arquiteturas, métodos e técnicas capazes de promover a adaptação de hiperdocumentos e hipermídia em geral às expectativas, necessidades, preferências e desejos de seus usuários. No presente capítulo abordam-se as perspectivas da apresentação e da navegação adaptativas, introduz-se o conceito de modelagem do usuário e se revisam algumas das principais técnicas de HA disponíveis para apresentação e suporte à navegação.

3.1 Conceitos Básicos

Sistemas de Hipermídia Adaptativa (HA) tentam antecipar as necessidades, preferências e desejos de seus usuários a partir de modelos representando seu perfil, nível de conhecimento, preferências, etc. Não há ainda um conceito plenamente estabelecido de HA, embora algumas tentativas de defini-la possam ser encontradas na literatura. Em [BRU 96], por exemplo, tem-se:

“(...) denomina-se sistema de hipermídia adaptativa todo sistema de hipertexto e hipermídia que reflita algumas características de seus diferentes usuários em modelos e aplique tais modelos na adaptação de diversos aspectos visíveis do sistema às necessidades e desejos de cada usuário”.

O objetivo geral dos sistemas e modelos de HA é portanto prover seus usuários com informação atualizada, subjetivamente interessante, com a ilustração multimídia pertinente, num tamanho e profundidade adequados ao contexto e em correspondência direta com o *modelo do usuário*. Este funciona como uma referência para o sistema, que busca adaptar seu ambiente - um hiperespaço, por vezes caótico - às expectativas particulares de seus usuários.

Os sistemas de HA devem então satisfazer a três critérios básicos: (1) ser um sistema hipertexto ou hipermídia, (2) possuir um modelo do usuário, e (3) ser capaz de adaptar a hipermídia do sistema usando tal modelo. Uma visão do laço de adaptação do sistema ao modelo do usuário é apresentada na Figura 3.1.

Sistemas de HA são especialmente úteis quando há a necessidade de disponibilizar informação seletiva e contextual a usuários com diferentes objetivos e níveis de conhecimento. Entre os principais usos da HA encontram-se hoje os sistemas educacionais baseados em hipermídia, sistemas de informações pessoais, sistemas de ajuda on-line, sistemas de informações institucionais e a construção de visões personalizadas. Novas aplicações começam agora a ser delineadas nas áreas de serviços personalizados, marketing dirigido, pesquisa de opinião, agendas coletivas, comunicação pessoal, etc.

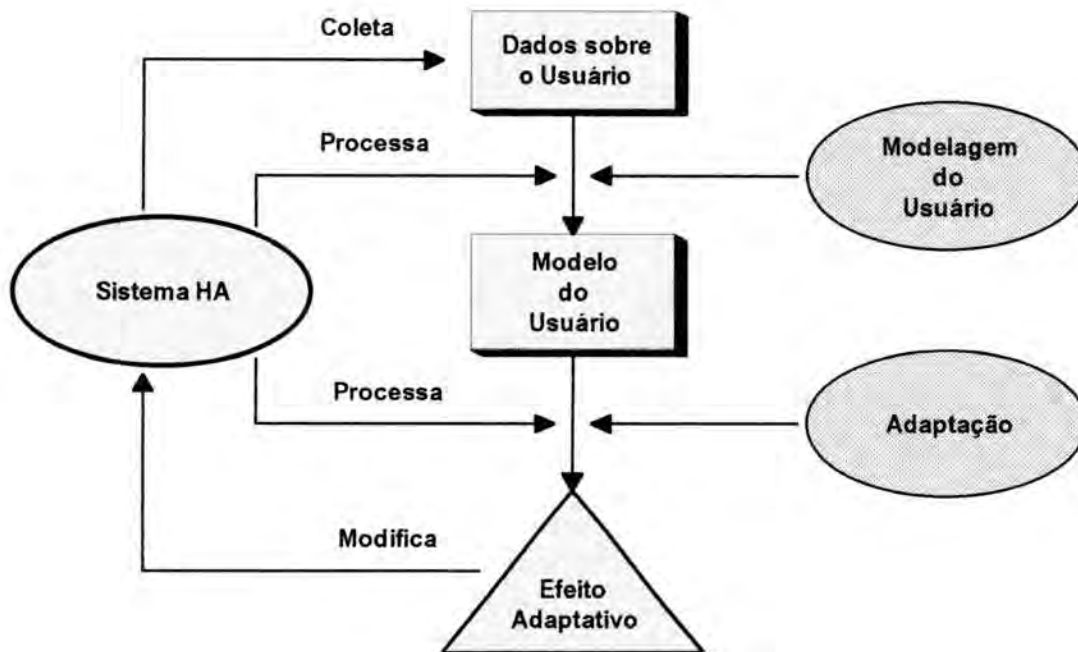


FIGURA 3.1 - O loop clássico: "modelo do usuário - adaptação"

Em se tratando de HA clássica, observa-se que o *espaço* de adaptação é bastante limitado. Não são muitas as características do sistema que podem ser alteradas no atual estágio tecnológico. Como foi visto no capítulo 2, em um certo nível de generalização o sistema hipermídia é constituído por um conjunto de nodos ou hiperdocumentos conectados por links. Cada nodo contém alguma informação local e links para outros nodos relacionados. Os sistemas hipermídia podem também incluir um *índice* ou um *mapa* com links para todos os nodos disponíveis. Nesta situação a adaptação pode ocorrer ao nível do conteúdo dos nodos ou ao nível dos links, índices e mapas. Estes dois níveis representam duas classes diferentes de HA, caracterizando o primeiro a *apresentação adaptativa* e o segundo a *navegação adaptativa*. [BRU 96]. Na Figura 3.2 apresenta-se uma classificação dos espaços da HA.

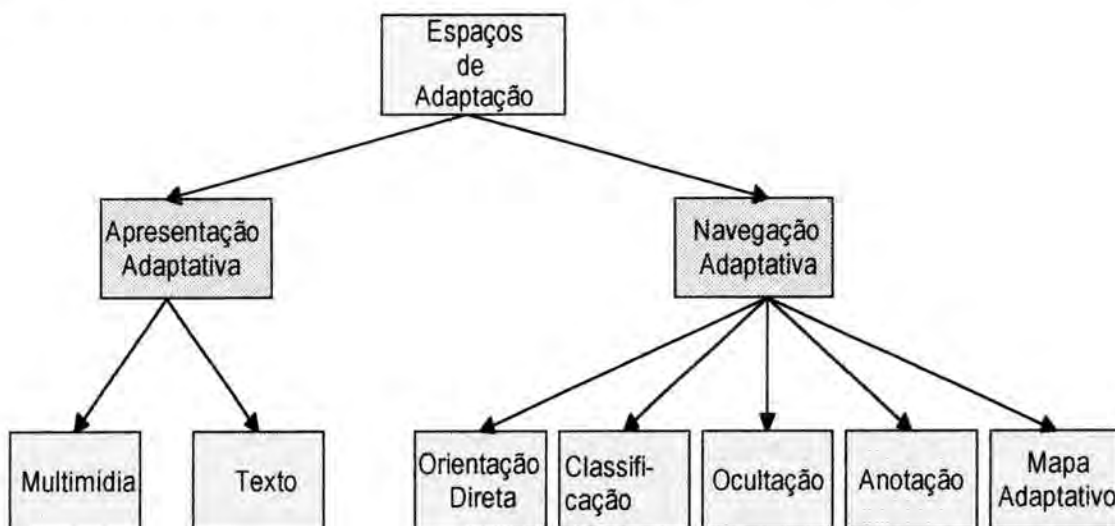


FIGURA 3.2 - Espaços de Adaptação em HA Clássica

3.2 Modelagem do Usuário

Há pelo menos cinco características associadas a um usuário (ou *papel* de um usuário) que podem ser levadas em conta por um sistema adaptativo: (1) conhecimento, (2) objetivos, (3) história, (4) experiência, e (5) preferências. Estas características são todas de algum modo *dinâmicas*, tornando necessário ajustar continuamente o modelo do usuário para garantir sua permanente atualização. A seguir comenta-se brevemente cada uma delas.

- *Conhecimento*: O conhecimento do usuário sobre o assunto representado no hiperespaço parece ser considerada a característica mais importante na construção do seu modelo. Segundo Brusilovsky [BRU 96], quase todas as técnicas de apresentação adaptativa utilizam o conhecimento do usuário como fonte de adaptação. Como este é dinâmico, isto é, se modifica ao longo do tempo para um particular usuário, o sistema deve ser capaz de reconhecer as modificações produzidas no conhecimento do usuário para atualizar adequadamente o seu modelo. As principais técnicas adotadas para a modelagem do conhecimento do usuário são a *sobreposição conceitual* [HOH 96] e *estereótipos* [BOY 94].
- *Objetivos*: Os objetivos do usuário estão mais relacionados com o trabalho ou atividade do usuário com hiperídia do que com ele próprio como indivíduo. Dependendo do tipo de sistema tais objetivos podem ser objetivos de trabalho (em sistemas aplicativos), objetivos de pesquisa (em sistemas de recuperação de informações) e a solução de problemas ou metas de aprendizado (em sistemas educacionais). Em todos esses casos o objetivo é a resposta à questão: “*Porque o usuário está usando o sistema hiperídia e o que exatamente deseja obter dele?*” Os objetivos do usuário são a sua característica mais sujeita a mudanças. Quase sempre se modifica de uma sessão para outra e freqüentemente pode mudar diversas vezes dentro de uma mesma sessão. Uma representação interessante usada para modelar os objetivos do usuário é obtida a partir de pares “*objetivo-valor*”, onde *valor* é normalmente a probabilidade de que o objetivo correspondente seja o objetivo *corrente* do usuário [MIC 96].
- *História e Experiência*: Dois outros fatores importantes que são semelhantes ao conhecimento na produção do modelo do usuário, mas que dele diferem funcionalmente, são a história do usuário e sua experiência anterior no hiperespaço considerado. Por *história* do usuário se quer significar toda informação relacionada com usos anteriores do sistema pelo usuário, fora do assunto abordado pelo sistema hiperídia, que sejam suficientemente relevantes para serem considerados. Já por *experiência* do usuário se quer denotar a familiaridade do usuário com a estrutura e navegação no hiperespaço considerado. Segundo Vassileva, isto não é a mesma coisa que o conhecimento do usuário acerca do assunto considerado. Usuários com pleno conhecimento do assunto tratado no hiperespaço podem ser totalmente ignorantes quanto à sua estrutura e, vice-versa, também é possível que o usuário domine perfeitamente a estrutura do hiperespaço, sem entretanto possuir qualquer conhecimento do assunto ali tratado [VAS 96].
- *Preferências*: Por diversas razões o usuário pode preferir certas associações de links e/ou conteúdos ao invés de outras que poderiam ser indicadas por

outras características de seu modelo. Tais preferências podem ser absolutas ou relativas, dependentes do nodo corrente, objetivos e contexto em geral. As preferências do usuário diferem das demais características componentes de seu modelo em diversos aspectos. Em geral as preferências não podem ser deduzidas pelo sistema. O usuário precisa declará-las ou informá-las indiretamente por meio de algum feedback simples. A principal diferença reside aqui no fato que os sistemas de hipermídia adaptativa devem ser capazes de generalizar as preferências do usuário e aplicá-las para adaptação a novos contextos [HÖÖ 98] [MAT 96]. A representação das preferências do usuário sob alguma forma numérica tem a vantagem de permitir a combinação de diversos modelos de diferentes usuários, por exemplo para acumular as preferências de um grupo específico de usuários (tal como um grupo de pesquisa em uma universidade, por exemplo). Um modelo de grupo poderia ser um excelente modelo inicial para um novo membro recém chegado ao grupo. Modelos de grupos são também importantes para o trabalho colaborativo, quando um modelo único (conjuntamente construído pelo grupo) pode ser muito mais eficiente e objetivo do que o possível uso de *brokers* (agentes de negociação) para intermediar a comunicação entre diversas visões adaptadas do mesmo assunto.

Na opinião de Brusilovsky [BRU 96], há alguns problemas a considerar na modelagem automática do usuário. Em primeiro lugar, esta não é considerada completamente confiável. Conforme apontado por [DeR 93], sistemas que executam a modelagem e a adaptação *sem* a influência do usuário são duplamente suspeitos. Podem cometer erros na dedução do modelo do usuário e podem errar também na execução da função de adaptação (mesmo que o modelo subjacente esteja correto). Em segundo lugar, alguns componentes do modelo do usuário, tais como sua história e preferências, não podem em absoluto ser deduzidos e precisam ser fornecidos diretamente pelo usuário. Por outro lado, quando complementadas por feedback ou intervenção direta do usuário, as técnicas de modelagem automática podem vir a ser muito eficientes.

No caso específico da hipermídia adaptativa, entretanto, a observação da ação do usuário não oferece elementos suficientes para a sua modelagem. Segundo Vassileva, as únicas informações que podem ser registradas sobre a ação do usuário são o caminho percorrido no hiperespaço e o tempo gasto em cada nodo [VAS 96]. Tais informações, embora interessantes, seriam insuficientes para a produção de um modelo confiável de um usuário em particular, embora possam muito bem ser consideradas para a modelagem de grupos e populações. Neste sentido, a pesquisa em sistemas complexos adaptativos [KAU 95] [MAY 95] [HEY 96] oferece evidências de que, a nível global e com pequeno investimento, grandes redes de informações poderiam se beneficiar da aplicação das técnicas de HA, capazes de produzir nelas um efeito organizador que poderia ser gerado a partir da observação do comportamento local de cada usuário. Na modelagem em grande escala, desvios locais tendem a desaparecer, restando apenas as conexões de maior interesse geral. Em contrapartida, a modelagem do usuário individual é tarefa bem mais complexa. Os múltiplos aspectos envolvidos no processo (modelagem estática, dinâmica, comportamental, etc.) contribuem para dificultar ainda mais sua execução, mantendo os resultados obtidos em níveis apenas parciais e restritos.

3.3 Métodos e Técnicas de Adaptação

Os sistemas clássicos de HA distribuem-se entre seis grandes áreas de aplicação: (1) recuperação de informações, (2) sistemas de informações, (3) sistemas de ajuda (help) online, (4) sistemas educacionais, (5) hipermídia institucional e (6) personalização de visões em espaços de informações [BRU 98]. Diversos sistemas baseados em HA surgidos a partir do início dos anos 90, implementaram uma grande variedade de métodos e técnicas de adaptação, algumas das quais são abordadas na presente seção.

3.3.1 Métodos de Apresentação Adaptativa

A idéia principal da apresentação adaptativa é adaptar o conteúdo de um nodo acessado por um particular usuário, ao conhecimento, objetivos e outras características deste usuário. Por exemplo, a um usuário qualificado é possível a apresentação de informação mais profunda e detalhada, enquanto que a um iniciante podem ser oferecidas explicações adicionais. É possível encontrar na literatura a descrição de técnicas e sistemas orientados a apresentação adaptativa de textos [STA 97] e de técnicas e sistemas orientados à apresentação adaptativa de objetos multimídia [BRU 97]. No caso de textos, o espaço de adaptação corresponde às possíveis modificações a que estes podem ser submetidos antes de serem apresentados ao usuário. Em geral isto se dá em nível de *recortes* e *atributos* de texto, que são adaptativamente selecionados para apresentação. Na apresentação de objetos multimídia o que costuma ocorrer é também a seleção dos objetos a serem apresentados a partir de um certo número de opções possíveis.

A *Explicação Adicional* (EA) é um dos métodos mais populares de adaptação de conteúdos e seu objetivo é ocultar do usuário alguma parte da informação sobre um certo conceito que não é relevante para o nível de conhecimento ou o interesse do usuário. Por exemplo, detalhes de baixo nível podem ser escondidos de usuários com nível de conhecimento insuficiente para entendê-los. Por outro lado, usuários novatos irão normalmente solicitar EA (para entender melhor certos conceitos) que pode permanecer oculta para usuários veteranos, que não necessitam mais dela. Em termos mais gerais, uma certa classe de usuários pode requerer informações adicionais, especialmente preparadas para eles, que não são mostradas às outras classes. Este método é usado por exemplo nos sistemas MetaDoc [BOY 94] e KN-AHS [KOB 94]. Uma variante deste método é ocultar do usuário certas partes da informação que não são relevantes para os objetivos do usuário [HÖÖ 98].

Dois outros métodos, denominados *Explicação Requerida* (ER) e *Explicação Comparativa* (EC), podem ser empregados para modificar a informação apresentada ao usuário dependendo do seu nível de conhecimento sobre os conceitos relacionados. O primeiro método induz uma ordenação dos conteúdos ao usuário onde a informação apresentada em primeiro lugar é pré-requisito para a seguinte. Seguindo esta idéia, ao apresentar a explicação de um conceito, o sistema insere a explicação de todos os conceitos requeridos para o seu entendimento. Este método é empregado no sistema C-book [KAY 94]. O método das EC baseia-se na similaridade existente entre dois conceitos. Se um conceito similar ao conceito que está sendo apresentado é conhecido, o usuário recebe uma explicação comparativa realçando as semelhanças e diferenças entre os dois conceitos. Este método costuma ser particularmente eficiente no domínio das linguagens de programação [BRU 98].

Um quarto método, denominado *Explicação Variante* (EV), assume que mostrar ou esconder certas partes da informação nem sempre é suficiente para promover a adaptação uma vez que usuários diferentes podem necessitar informação essencialmente diferente. Através deste método o sistema armazena diversas variantes para alguns dos conteúdos de uma página e o usuário obtém a apresentação da EV que corresponde ao seu modelo. Este método é empregado em Anatom-Tutor [BEA 98] e Hypadapter [HOH 96].

Um quinto método ainda, bastante interessante, que leva em conta o nível de conhecimento e a experiência do usuário é a *Classificação de Fragmentos* (CF), que ordena fragmentos de informação sobre o conceito de modo que a informação mais relevante para o usuário (de acordo com o seu modelo) é apresentada primeiro. Este método é implementado pelos sistemas Hypadapter [HOH 96] e EPIAIM [DeR 93].

3.3.2 Técnicas de Apresentação Adaptativa.

Uma técnica simples, porém eficiente, para a apresentação adaptativa é a do Texto Condicional (TC), utilizada nos sistemas ITEM-IP [BRU 92], Lisp-Critic [FIS 90] e C-book [KAY 94]. Com TC é possível dividir a informação em diversas porções de texto. Cada porção é associada a uma ou mais condições relacionadas ao nível de conhecimento do usuário. Ao apresentar a informação o sistema mostra apenas as porções de texto que tiveram suas condições satisfeitas. Esta é uma técnica de baixo nível, que requer alguma programação, entretanto é também muito flexível e permite implementar todos os métodos de adaptação relacionados acima, à exceção da Classificação de Fragmentos. Um exemplo simples é a ocultação de porções de informação que não são relevantes para o nível de conhecimento do usuário, ou a apresentação de uma explicação comparativa se o conceito relacionado já é conhecido.

Uma técnica de nível mais alto, que também permite apresentar ou ocultar condicionalmente porções de texto de acordo com o nível de conhecimento do usuário, foi sugerida no sistema MetaDoc [BOY 94] e posteriormente desenvolvida em KN-AHS [KOB 94]. Esta técnica baseia-se no *stretchtext* que é um tipo especial de hipertexto onde os links podem ser expandidos para seus conteúdos ou concentrados novamente em uma palavra chave. A idéia aqui é apresentar ao usuário uma página onde todas as informações relevantes estejam expandidas e todas as informações irrelevantes sejam representadas por apenas uma palavra ou frase, de acordo com o modelo de seu particular conhecimento.

O método EV pode ser implementado pelas técnicas de *Fragmento Variante* (FV) ou *Página Variante* (PV). Esta última é considerada a mais simples das técnicas de apresentação adaptativa e consiste em manter duas ou mais páginas alternativas para cada conceito, descrevendo-o de maneiras diferentes, cada uma delas adaptada a uma certa classe de usuário. Esta técnica é empregada nos sistemas Anatom-Tutor [BEA 98] e C-book [KAY 94]. Uma técnica similar foi empregada em EPIAIM [DeR 93] para adaptar a apresentação de exemplos à experiência do usuário. Este sistema armazena diversos exemplos sobre cada conceito e apresenta ao usuário aquele que melhor se adapta aos seus interesses e experiência anterior.

A técnica FV permite a implementação do método EV em uma granulação mais fina do que a técnica PV. Um bom exemplo aqui é o sistema Anatom-Tutor [BEA 98], onde não há uma correspondência entre uma página e um conceito, como acontece em outros sistemas. Ao invés disso, uma página pode apresentar vários conceitos. Cada

conceito por sua vez pode possuir diversos fragmentos variantes e a página é instanciada com a combinação deles que melhor satisfaz aos conhecimentos do usuário. O sistema Anatom-Tutor combina as técnicas PV e FV de modo a produzir adaptação simultânea à experiência e ao conhecimento de seus usuários. A página variante corrente para um determinado nodo é selecionada de acordo com a *experiência* do usuário e pode ser ainda configurada com fragmentos variantes selecionados para cada um dos conceitos expressos na página, visando a adaptação ao seu *conhecimento*.

A mais eficiente de todas as técnicas de apresentação adaptativa é sem dúvida a representação por meio de frames, que pode ser observado em Hypadapter [HOH 96] e EPIAIM [DeR 93]. Por meio desta técnica, toda a informação sobre um determinado conceito é representada sob a forma de um *frame*. Estes são estruturas de atributos e valores a eles associados, armazenados em *slots*, que descrevem uma entidade do mundo [RIC 94]. Os slots do frame podem conter diversas EV sobre o conceito, links para outros frames, exemplos, etc. Regras especiais de apresentação são empregadas para decidir quais os slots de um determinado frame devem ser apresentados a um certo usuário e em que ordem específica isto deve ocorrer. Mais precisamente, no sistema EPIAIM estas regras são empregadas para selecionar um dentre vários esquemas de apresentação disponíveis (onde cada esquema é um conjunto ordenado de slots) que é usado para a apresentação do conceito. No sistema Hypadapter, as regras são empregadas para calcular a prioridade de apresentação de cada slot. Em seguida o subconjunto de slots com maior prioridade é apresentado ao usuário.

Outra técnica muito eficiente para a apresentação adaptativa foi desenvolvida no projeto PUSH [HÖÖ 98] e pode ser considerada uma combinação da técnica stretchtext com a adaptação por meio de frames. Uma página hipermídia neste sistema de informações online oferece a descrição completa de um objeto particular, estruturada em uma seqüência ordenada de porções tipadas de informação. Cada classe de objeto no sistema PUSH possui a sua própria ordenação dos tipos de informação que são empregados na sua descrição. A tipagem da informação desempenha um papel muito próximo do executado pelos slots na representação por meio de frames, permitindo descrever os diversos aspectos dos objetos representados. A descrição de um objeto neste sistema entretanto costuma ser muito grande, ocupando diversas páginas de hipertexto.

Para proteger o usuário do excesso de informações e auxiliá-lo a atingir a informação desejada o sistema emprega o stretchtext, mostrando apenas as porções de informação sobre o objeto corrente que são relevantes para os objetivos do usuário. É permitido ao usuário modificar a apresentação do sistema, expandindo ou contraindo as porções de informação apresentadas. Entretanto, para manter a transparência da adaptação, a ordenação das porções de informação é mantida estável e o sistema nunca as oculta completamente, nem mesmo quando são consideradas irrelevantes para os objetivos do usuário, mantendo sempre visíveis ao menos os seus títulos, de modo que o usuário sempre tem acesso a todas as informações.

Na tabela a seguir apresenta-se a associação dos diferentes métodos e técnicas descritos na presente seção com os sistemas citados que as implementam.

TABELA 3.1 - Métodos e técnicas de apresentação adaptativa em sistemas de HA

| Técnicas Métodos | Texto Condicional | Stretchtext | Fragmentos Variantes | Páginas Variantes | Frames |
|--|----------------------------------|---------------------------|-----------------------------|----------------------------------|----------------------|
| Explicações Adicionais, Requeridas e Comparativas | C-book ITEM/IP Lisp-Critic | MetaDoc KN-AHS PUSH | | | EPIAIM PUSH |
| Explicações Variantes | C-book | | Anatom-Tutor Lisp-Critic | Anatom-Tutor C-book EPIAIM | Hypadapter |
| Classificação | | | | | EPIAIM Hypadapter |

3.3.3 Métodos de Navegação Adaptativa

O objetivo da navegação adaptativa é auxiliar os usuários a encontrar seus caminhos no hiperespaço através da adaptação da forma de apresentar os links na rede hipermídia aos objetivos, conhecimento e outras características de seus usuários. Os métodos de suporte à navegação podem ser resumidos nos seguintes:

1. Condução Global;
2. Condução Local;
3. Suporte à Orientação Local (Conhecimento);
4. Suporte à Orientação Local (Objetivos);
5. Suporte à Orientação Global.

A *Condução Global* (CG) ocorre quando os usuários possuem algum objetivo global de informação que se encontra em um ou mais nodos que estão em algum lugar do hiperespaço e são "conduzidas" pelo sistema nesta direção. O objetivo deste método é ajudar o usuário a encontrar o caminho mais curto para a informação que ele deseja, com possíveis desvios minimizados. A condução global é o objetivo primário do suporte à navegação adaptativa em sistemas de recuperação de informações hipermídia e também importante em sistemas de informações e sistemas de ajuda online. O objetivo de informação do usuário é usualmente fornecido de modo mais ou menos claro pelo próprio, que é o principal agente do processo. O método mais direto de oferecer condução global é sugerir ao usuário em cada passo da navegação os links mais apropriados para atingir a partir do nodo corrente a informação desejada. Este método é empregado no sistema WebWatcher [ARM 95]. Um método complementar é a classificação dos links que serão sugeridos ao usuário em ordem decrescente da relevância que possuem para os objetivos do usuário. Os sistemas Adaptive HyperMan [MAT 96] e HYPERFLEX [KAP 93] implementam este método.

Um caso especial de CG ocorre nos sistemas educacionais, onde o estudante tem um objetivo global, que é o objetivo do aprendizado. Este objetivo é diferente dos objetivos de informação, tendo em vista diversos fatores como a especificidade e profundidade do conhecimento em relação à informação. Sistemas educacionais adaptativos necessitam capturar a dinâmica do aprendizado em cada um dos seus usuários e providenciar a condução global tendo em vista esta dinâmica, retardando ou acelerando tópicos em função do feedback fornecido pelo usuário.

O objetivo da *Condução Local* (CL) é semelhante ao da CG, porém de alcance muito menor. Enquanto que a CG preocupa-se com seqüências de links que conduzem ao objetivo desejado, a CL ocupa-se de um único passo e tenta sugerir ao usuário os links mais relevantes considerando suas preferências, conhecimento e experiência. Por exemplo, um método para condução local é classificar os links de acordo com as preferências do usuário, como é feito nos sistemas Adaptive HyperMan e HYPERFLEX. Os métodos empregados em sistemas educacionais adaptativos são a classificação dos links e a condução direta de acordo com o conhecimento do usuário.

O *Suporte à Orientação Local* (OL) consiste em auxiliar o usuário a entender o seu posicionamento na rede do hipertexto local, correspondente ao domínio do sistema. Isto é normalmente implementado em sistemas de HA de duas formas distintas: por meio de informação adicional sobre os nodos, que podem ser acessados a partir do nodo corrente, ou limitando as oportunidades de navegação, visando com isso evitar o fenômeno denominado *sobrecarga cognitiva*, onde o excesso de informação impede ou dificulta a tomada segura de decisão na navegação. Tais métodos são em geral baseados na técnica da *ocultação*, que se concentra em remover das vistas do usuário toda a informação que não é relevante para os seus objetivos num dado momento. O método considerado mais simples nesta categoria é mostrar somente os links relevantes para as preferências estabelecidas dos usuários [WAT 96]. O mais universalmente empregado entretanto é o que considera o objetivo corrente do usuário. Este método foi empregado nos sistemas HyPlan [GRU 93], Hynecosum [VAS 96] e PUSH [HÖÖ 98]. Um outro método, adotado no sistema HyperTutor [PÉR 95], consiste em mostrar mais links para os usuários com maior experiência. Os usuários novatos vêem somente um pequeno número de links, que vai crescendo na medida em que sua experiência aumenta.

Dois métodos de OL são específicos para hipermídia educacional. Um deles, muito popular, consiste na ocultação dos nodos que o usuário não está preparado para aprender. Este método é empregado em diversos sistemas: ISIS Tutor [BRU 94], HyperTutor [PÉR 95] e Hypadapter [HOH 96]. Outro método, específico para sistemas educacionais adaptativos consiste em esconder os links para nodos que pertencem a objetivos educacionais de outras lições e estão fora dos objetivos da presente lição. Isto é feito nos sistemas ISIS Tutor e SYPROS [GON 95].

A idéia de métodos de adaptação baseados em *anotação* é informar o usuário acerca do estado corrente dos nodos, além do que é visível nos links explícitos. São quatro os métodos mais utilizados nesta categoria. Primeiro a anotação pode ser usada para apresentar as diversas graduações da relevância dos nodos para os objetivos do usuário. Por exemplo, o sistema Hypadapter possui três diferentes tamanhos de fonte indicando os nodos que são muito relevantes, relevantes ou menos relevantes. O segundo método consiste em refletir os diversos níveis de conhecimento do usuário acerca dos nodos [SCE 96]. Os dois outros métodos empregam anotação onde

usualmente é usada a ocultação. Um deles consiste em *sublinhar* os links que estão diretamente relacionados com o objetivo corrente [BRU 94]. A outra é oferecer anotação especial para os links que o usuário ainda não está preparado para aprender, como nos sistemas ISIS-Tutor e ELM-ART [BRU 94].

O objetivo da *Orientação Global* (OG) é ajudar o usuário a entender a estrutura de todo o hiperespaço que constitui o domínio de navegação do sistema. Em sistemas não-adaptativos isto é obtido por meio de marcos visuais e mapas globais que auxiliam o usuário a localizar sua posição em relação ao contexto global [LIN 95]. Sistemas de HA podem oferecer um suporte maior nesta mesma linha pela aplicação das técnicas de ocultação e anotação, o que pode ocorrer de forma sistemática, contemplando o modelo do usuário independentemente da sua posição no hiperespaço. O método mais amplamente utilizado aqui é aumentar gradualmente o número de links visíveis na medida em que vai crescendo a experiência do usuário no hiperespaço considerado. Isto ocorre nos sistemas HyperTutor e Hynecosum [VAS 96].

Um objetivo mais recente para os sistemas de HA é o gerenciamento de visões personalizadas do hiperespaço, uma forma de construir interfaces de trabalho personalizadas por meio de adaptação. Estas visões são necessárias em ambientes dinâmicos na web, onde links podem aparecer, desaparecer e evoluir. Um pioneiro na aplicação deste método é o sistema BASAR [THO 96], que emprega agentes inteligentes para coletar e manter um conjunto de links relevantes para cada um dos seus usuários. Estes agentes executam pesquisas regulares na web por novos links relevantes para cada usuário e verificam a expiração ou atualização ocorridas nos links coletados.

3.3.4 Técnicas de Navegação Adaptativa

A noção de link que se adota aqui é a que se apresenta do ponto de vista do usuário, isto é, a representação visível e “clicável” dos nodos relacionados pelos quais o usuário pode navegar. Em [BRU 96] os links são classificados segundo quatro diferentes classes:

1. *Links locais, não contextuais*: Correspondem aos links independentes do conteúdo do nodo em que se encontram, normalmente apresentados como um conjunto de botões, uma lista ou um menu pop-up. Tais links são fáceis de manipular e podem ser classificados, ocultos ou anotados.
2. *Links contextuais ou “hipertexto verdadeiro”*: Compreendem os links vinculados ao contexto, tais como os representados por certas palavras ou frases em um texto ou por zonas especialmente delimitadas de uma imagem. Estes links podem ser anotados, mas não podem ser classificados nem totalmente ocultos.
3. *Links para índices e tabelas de conteúdos*: Um nodo com índices ou tabelas de conteúdo podem ser vistos como um tipo especial de nodo que só contém links. Tais links são assumidos como não-contextuais, a menos que sejam implementados sob a forma de uma imagem.
4. *Links para mapas locais e globais*: Mapas são representações gráficas de um hiperespaço ou área local de um hiperespaço como uma rede de nodos conectados por setas. O usuário pode navegar diretamente sobre todos os nodos visíveis no mapa.

A partir desta classificação dos links em uma rede hipermidia, é possível comparar as técnicas existentes para a sua adaptação. A *Orientação Direta* (OD) é talvez a mais simples das técnicas de suporte à navegação adaptativa e sua aplicação consiste em decidir, em cada ponto da navegação, qual o *melhor* nodo a ser visitado a seguir, levando em conta os objetivos, preferências, conhecimento e outros parâmetros representados no modelo do usuário. Para oferecer orientação direta o sistema pode destacar visualmente o link para o *melhor* nodo, como é feito no WebWatcher [ARM 95] ou apresentar um link dinâmico adicional (normalmente denominado “next”) que é conectado ao melhor nodo selecionado, como no sistema ISIS-Tutor [BRU 94]. A orientação direta é uma técnica clara e fácil de implementar, que pode ser empregada em todas as quatro classes de links anteriormente citadas. O problema é que em geral ela não oferece suporte aos usuários que não quiserem seguir a sugestão do sistema, o que é um indicativo de que, apesar de útil, esta técnica não deve ser empregada sozinha, como único recurso de adaptação em um sistema.

A técnica de *Classificação Adaptativa* (CA) consiste em classificar todos os links partindo de um nodo de acordo com a sua *relevância*, calculada sobre o modelo do usuário. Os links são apresentados então em ordem decrescente desta relevância. Segundo [BRU 96] a classificação adaptativa possui uma aplicação limitada: pode ser empregada satisfatoriamente com links não-contextuais, mas seu uso com índices e tabelas de conteúdos é muito difícil e a técnica não pode ser absolutamente usada para links contextuais nem para mapas. Entretanto esta técnica parece ser útil em recuperação de informações [ARM 95] [MAT 96]. Resultados empíricos [KAP 93] mostraram que a classificação adaptativa pode reduzir significativamente o tempo de navegação em aplicações de recuperação de informações, onde cada nodo pode apresentar muitos links não-contextuais. A classificação adaptativa pode ainda ser empregada em sistemas de documentação on-line [HOH 96] e em hipermidia educacional [BRU 96].

A técnica de *Ocultação* é a mais frequentemente empregada em navegação adaptativa e consiste em restringir o espaço de navegação ocultando os links para nodos não relevantes. Um nodo pode ser considerado não-relevante por diversas razões, por exemplo, se não está relacionado com o objetivo corrente do usuário, ou este não possui conhecimento suficiente para entendê-lo. De um ponto de vista superficial a ocultação parece ser a técnica mais óbvia e fácil de implementar. Seu uso protege o usuário da complexidade de um hiperespaço irrestrito e reduz a sua sobrecarga cognitiva. A ocultação possui ampla aplicação, podendo ser empregada com todos os tipos de links. Com links não-contextuais, índices e mapas a ocultação se dá tornando invisíveis ao usuário botões, itens de menu e áreas *clicáveis* irrelevantes aos seus objetivos. Com links contextuais a técnica é eliminar a diferenciação entre as palavras associadas a links e o texto comum. A ocultação é também mais transparente ao usuário e produz uma apresentação mais estável do que a classificação adaptativa.

A idéia da *Anotação Adaptativa* (AA) é aumentar a informação presente nos links com alguma forma de anotação ou comentário que podem dizer mais sobre o estado corrente dos nodos a que se conectam. Esta informação adicional pode ser oferecida sob a forma de texto ou sob a forma de indicadores visuais, tais como ícones especiais, cores ou tamanho dos caracteres. Em geral a anotação oferecida em hipermidia convencional é do tipo estático, isto é, independente do usuário. Em se tratando de HA, entretanto, tem-se a expectativa que a anotação seja dinâmica, orientada ao modelo do usuário.

A anotação pode ser usada naturalmente com todos os quatro tipos de links considerados. Seu uso mantém uma ordenação estável dos links e em geral é uma técnica mais poderosa do que a ocultação, na medida em que esta última oferece somente dois estados possíveis para os links (relevante/visível e não-relevante/oculto), enquanto que a anotação adaptativa pode estabelecer diversos *níveis de relevância* [HOH 96]. Além disso a anotação pode simular a ocultação simplesmente obscurecendo os itens considerados não-relevantes. O obscurecimento pode reduzir em alguma extensão a sobrecarga cognitiva (o usuário pode aprender a ignorar os links obscurecidos) mas estes se manteriam ainda visíveis (e operacionais, se for necessário).

A técnica dos *Mapas Adaptativos* (MA) compreende diversas formas de adaptação de mapas de hipermídia global e local apresentados ao usuário. Como se viu, as técnicas de orientação direta, ocultação e anotação podem também ser usadas na adaptação de mapas hipermídia, entretanto o emprego de tais técnicas não modifica a forma ou a estrutura dos mapas. A pesquisa realizada na área da interação homem-máquina oferece diversas técnicas para a adaptação da forma e estrutura de diversos tipos de redes, incluindo mapas hipermídia [MUK 95]. Tais técnicas entretanto, não foram ainda aplicadas em sistemas de HA. A única exceção conhecida é o sistema Hypercase [MIC 96], parcialmente baseado nas idéias de Rivlin [RIV 94].

As técnicas de orientação direta, classificação, ocultação, anotação e de mapas adaptativos são as técnicas primárias empregadas na navegação adaptativa. Elas não são mutuamente exclusivas ou contraditórias, de modo que é perfeitamente possível o seu emprego combinado. Por exemplo o sistema ISIS-Tutor [BRU 94] emprega orientação direta, ocultação e anotação e o Hypadapter [HOH 96] usa classificação, ocultação e anotação. Em particular a orientação direta pode ser naturalmente empregada em combinação com qualquer outra técnica. Na Tabela 3.2, a seguir apresenta-se o relacionamento existente entre os métodos, técnicas e sistemas de HA sob o ponto de vista da navegação adaptativa.

Um elemento essencial em todos os métodos e técnicas apresentados no presente capítulo é a relação de relevância que pode existir entre dois nodos quaisquer do hipertexto e que se comporta de maneira dinâmica, surgindo e ganhando importância ou decaindo e desaparecendo. Este aspecto dinâmico tem impacto não somente na modelagem do usuário mas também na modelagem da hipermídia e em todo o processo de adaptação. O tratamento da questão da relevância é portanto uma preocupação básica de qualquer sistema de HA. Como decidir se um determinado nodo ou estrutura no hiperespaço é relevante para o nodo corrente, dado o conhecimento e/ou os objetivos do usuário? Nos próximos capítulos desenvolve-se um estudo de dois métodos candidatos a representar e processar dinamicamente esta relação. O primeiro segundo a abordagem lógica simbólica da Teoria das Situações e o segundo a linha conexionista através de um método associativo baseado na Teoria do Fecho.

TABELA 3.2 - Métodos e Técnicas de Navegação Adaptativa em Sistemas de HA

| Técnicas | Orientação Direta | Classificação | Ocultação | Anotação | Mapas Adaptativos |
|---------------------------------|------------------------------------|-------------------------------------|---|----------------------------------|-------------------|
| Métodos | | | | | |
| Condução Global | WebWatcher ITEM/IP HYPERFLEX | HyperMan HYPERFLEX | | | |
| Condução Local | HyperTutor | HyperMan Hypadapter HYPERFLEX | Hypadapter PUSH | ISIS-Tutor ELM-ART | Hypercase |
| Orientação Local - Conhecimento | | Hypadapter | HyperTutor Hypadapter ISIS-Tutor | ISIS-Tutor ELM-ART ITEM/PG | |
| Orientação Local - Objetivos | | | Hynecosum HyPlan ISIS-Tutor PUSH SYPROS | ISIS-Tutor ELM-ART ITEM/PG | Hypercase |
| Orientação Global | | | Hynecosum HyperTutor ISIS-Tutor SYPROS | ISIS-Tutor ELM-ART ITEM/PG | Hypercase |

Capítulo 4

Um Modelo Conexcionista para Hipermídia Adaptativa

Neste capítulo se propõe um modelo conexionista para suportar a adaptação da navegação em sistemas de hipermídia adaptativa. O modelo conexionista destaca a representação comportamental das ligações considerando a frequência com que os links são percorridos na rede de conexões. O processo é balizado por leis de transitividade e reflexividade, que associadas à teoria do fecho permitem modelar a evolução do hiperespaço através da evolução de seus links, abstraídos do conteúdo dos nodos. O modelo conexionista apresentado pode ser configurado para implementar diversos métodos de adaptação.

4.1 Introdução

Diversos estudos sobre complexidade em sistemas dinâmicos podem ser encontrados na literatura nas mais variadas áreas da ciência, incluindo a física, a química, a biologia, a psicologia, a sociologia e a informática [KAU 93] [PRI 84]. Estes estudos apontam que a ordem observada no universo vem do interior dos sistemas existentes, criada pela mesma dinâmica que regula o comportamento de seus componentes.

Sistemas de informações distribuídos, tais como a World Wide Web, podem ser vistos como *redes de informações*, que exibem uma certa forma de dinâmica onde a ordem surge dos relacionamentos intuitivos produzidos pelos usuários enquanto navegam através de seus nodos [PAL 97]. A hipótese assumida aqui é que este processo pode ser usado não somente como uma heurística direta para reduzir o espaço de pesquisa na recuperação de informações, mas também para oferecer as bases para a inferência de novos (e interessantes) caminhos em redes de informações.

A investigação dessa hipótese requer um modelo adequado para a representação e processamento dinâmico da informação. A abordagem proposta no presente capítulo emprega conceitos de redes complexas, estendidos com operadores simbólicos e regras de inferência para compor panoramas de redes com informação nova e potencialmente útil. O modelo subjacente possui duas visões integradas: a visão *adaptativa* e a visão *inferencial*. A visão adaptativa modela a dinâmica da navegação sobre a rede por meio de *links quantizados*. Estes se caracterizam por terem medido o seu *potencial de ativação*, isto é a sua probabilidade de serem ativados no próximo instante ou intervalo de tempo. Este valor de referência sobre os links em uma rede de informações permite a composição e a extração de sub-redes com características definidas pelo sistema a partir do total de possibilidades oferecidas pela rede.

O componente inferencial usa o conceito de fecho para criar novos links ou reforçar o potencial de ativação dos links existentes. Isto complementa o efeito adaptativo, levando as redes de informações a evoluir em direções novas e não consideradas anteriormente durante seu uso. No modelo aqui proposto, ambas as visões coexistem e a variedade oferecida pelo fecho é contrabalançada pela restrição imposta ao sistema pelo componente adaptativo. Em outras palavras, as redes resultantes devem

buscar adaptar-se ao processo de navegação que suportam, ao mesmo tempo em que tentam novos links promissores para manter-se em evolução. Uma importante questão aqui é a determinação precisa dos valores dos parâmetros a adotar para o melhor desempenho do sistema. Aparentemente tais valores são muito específicos e variam em diferentes situações, sendo portanto necessário aferi-los caso a caso.

4.2 Redes de Informação

Uma Rede de Informações é uma tripla, $\langle N, L, P \rangle$, onde:

- N é um conjunto de *nodos* que são entidades primitivas que correspondem a unidades estruturais, semânticas e referenciais do sistema. Um nodo possui seus próprios *atributos* tais como uma identidade, conteúdo, pertinência a classes, métodos, etc. Um nodo se relaciona com outros nodos por meio de *links*. Nodos que se encontram em relacionamento direto por meio de um link são ditos ser *vizinhos*. A *vizinhaça* de um nodo é esse nodo juntamente com todos os seus vizinhos.
- L é um conjunto de *links*, que são entidades estruturais estabelecendo a existência e definindo os detalhes de um relacionamento entre dois nodos. Os links podem apresentar diferentes semânticas, conforme os nodos e o sistema considerado. No presente modelo os links denotam um relacionamento abstrato que possui associado um *potencial de ativação*.
- P é um conjunto de propriedades que valem para a rede. Estas propriedades não são pré-definidas e podem ser diferentes em cada situação.

As redes de informações podem ser vistas então como *redes quantizadas*, onde os links possuem potenciais de ativação. Na Figura 4.1 apresentam-se dois estágios de **uma** rede de informações simples, onde os links foram quantizados no estágio 2 para representar o processo de navegação.

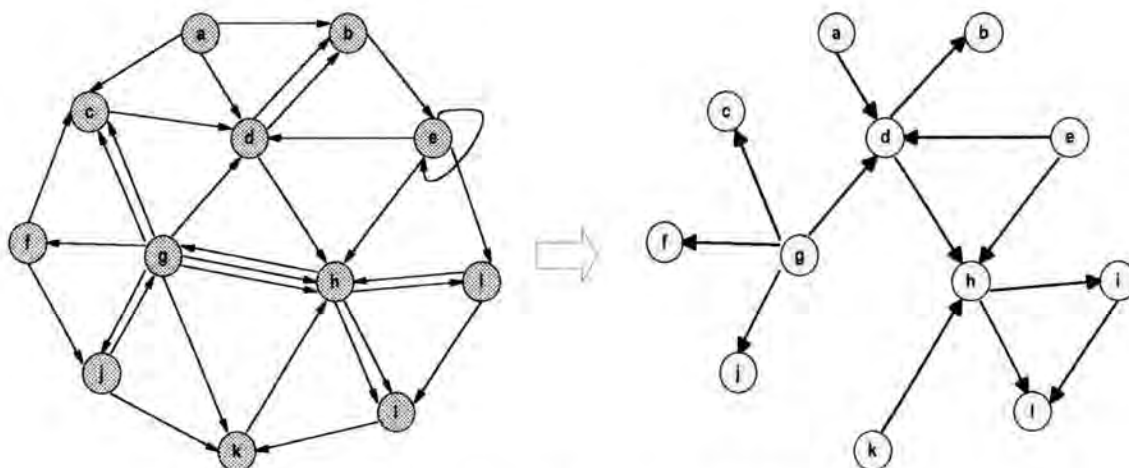


FIGURA 4.1 - Quantização em redes de informação

Na figura acima as setas em negrito no diagrama da direita mostram apenas os links que possuem potencial de ativação acima de um determinado valor denominado *limiar de visibilidade*. Este valor, que pode ser escolhido arbitrariamente, funciona como um seletor dos links que são visíveis ou não na rede. Na proposta aqui apresentada, redes quantizadas são empregadas na modelagem dinâmica de redes de

informações. Nestas redes, o potencial de ativação associado a cada link é incrementado de um valor arbitrário sempre que o link é percorrido. Por outro lado, se o link não é percorrido durante um certo intervalo de tempo, seu potencial de ativação decresce. Esta degradação do potencial de ativação com a passagem do tempo é imposta a toda a rede e seu efeito é de fundamental importância para manter a atualização do modelo.

4.3 Inferência em Redes através de Operadores de Fecho

Na seção anterior apresentou-se uma forma de adaptação *retroativa*, onde o modelo reflete uma dinâmica pretérita, já ocorrida sobre a rede. Em complementação a este processo, é possível desenvolver métodos de adaptação *proativa*, onde novos links são gerados por alguma forma de inferência. No modelo que se está propondo aqui empregam-se operadores de fechamento para *gerar* novos links com base na evolução da rede. A base teórica deste modelo vem da noção matemática de *fecho* [HEY 88].

A mais fundamental das propriedades do fecho é talvez a *internalidade de composição*, que significa que quaisquer duas transformações ou relações no sistema podem ser compostas seqüencialmente de modo a formar uma terceira transformação que é também parte do sistema original. Em outras palavras a internalidade de composição torna o sistema invariante sob a composição de transformações. Este tipo de fecho básico é denominado *fecho transitivo* ou *recursivo*.

Outro fecho básico é implicado pela existência de transformações *simétricas* ou *inversas*. Isto significa que o sistema é invariante sob a inversão de transformações. Na prática quer dizer que se for aplicada uma transformação a um elemento, pode-se sempre reverter o seu efeito através da aplicação da transformação inversa sobre o elemento resultante da transformação inicial - o que conduz de volta ao elemento original. Isto pode ser denominado um *fecho simétrico* e é um caso especial de um tipo de fecho mais geral, denominado *fecho cíclico* ou *circular*, que estabelece que qualquer seqüência de transformações pode ser invertida pela aplicação de outra seqüência específica de transformações ao seu resultado, de forma que a concatenação de ambas as seqüências forma um *caminho cíclico*. Deve-se notar que no fecho cíclico não é absolutamente necessário que a transformação inversa empregada seja composta pelas inversas das relações componentes da transformação original.

Implícita no conceito de fecho está a idéia de que uma transformação deve corresponder a um mapeamento *bijetor*, isto é, uma relação 1-1. Entretanto em geral uma relação pode ser n-1, 1-n ou n-n. A restrição de uma relação n-n para uma relação n-1 (sobrejetora) pode ser novamente entendida como um fecho sobre a estrutura relacional do sistema. Na verdade uma relação R do tipo n-1 é caracterizada pelo fato de que a composição com sua inversa R^{-1} produz uma identidade.

$$R[R^{-1}[a]] = a, \quad \forall a$$

de modo equivalente em relações 1-n tem-se:

$$R^{-1}[R[a]] = a, \quad \forall a$$

No caso de relações 1-1 ambas as propriedades se aplicam. Assim os elementos de entrada ou saída sobre os quais as relações se aplicam são invariantes sob as transformações $R \circ R^{-1}$ ou $R^{-1} \circ R$ e portanto podem ser considerados um sistema

fechado. Um mapeamento 1-1 pode ser caracterizado pelo fato de conservar todas as distinções entre os elementos: elementos distintos mapeiam elementos distintos.

As citadas propriedades são denominadas respectivamente fecho sobrejetor ($n-1$), fecho sobrejetor inverso ($1-n$) e fecho bijetor ($1-1$). Os quatro tipos de fecho sobre grafos de duas setas podem ser observados na figura 4.2 a seguir. A generalização de tais propriedades em âmbito local pode gerar interessantes resultados em nível global, configurando automaticamente a rede para um comportamento dinâmico adaptável em perseguição de índices ótimos ainda não atingidos em sua função de adequação.

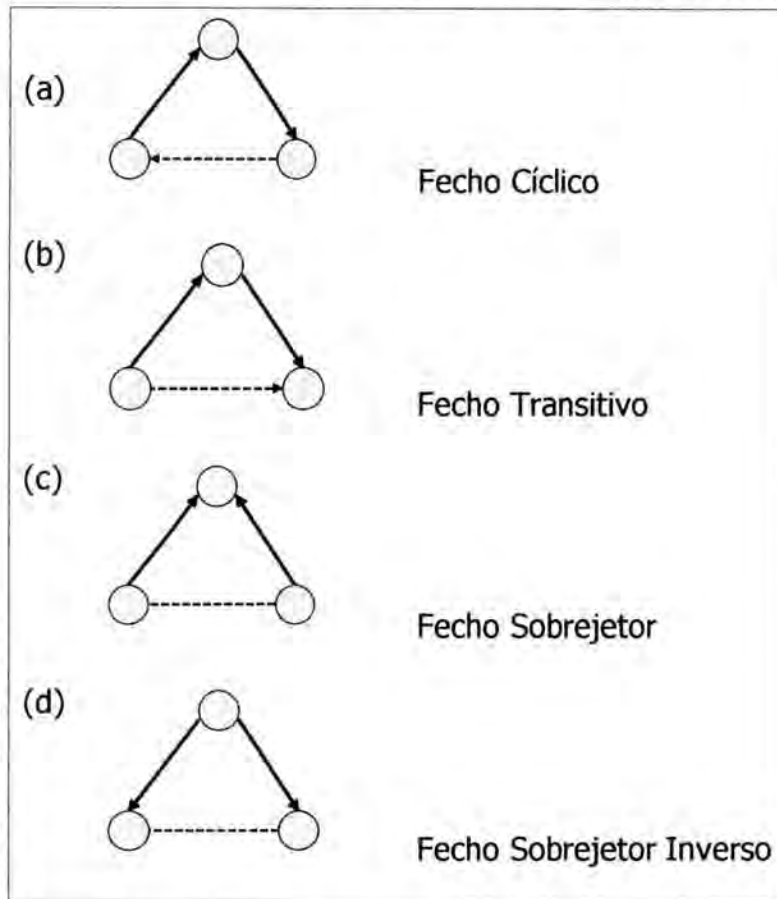


FIGURA 4.2 - Tipos de fechos em grafos de duas setas

Nos diagramas (a) e (b) da figura 4.2 as duas setas originais (em linha cheia) estão conectadas *seqüencialmente*, de forma que em relação ao nodo comum uma está chegando e a outra partindo. A terceira seta na base (em linha pontilhada), que é adicionada para formar o fecho, pode ser orientada de dois modos distintos: continuando a seqüência das duas setas originais, conduzindo de volta ao nodo inicial (fecho cíclico), ou em paralelo com a seqüência (fecho transitivo). Nos diagramas (c) e (d) as duas setas originais são conectadas em paralelo. Como o sistema formado nestes dois casos apresenta uma simetria geral, a conexão adicionada para formar o fecho não deve ser orientada (nenhuma seta na linha pontilhada). A simetria da conexão original pode ser orientada como uma relação de equivalência, conduzindo a identificação dos dois nodos conectados, de forma que eles não são mais distinguidos individualmente, mas somente como uma coleção. O resultado é que também as setas originais não são mais distinguidas. As relações correspondem então respectivamente ao fecho sobrejetor ($n-1$) e ao fecho sobrejetor inverso ($1-n$).

4.4 Adaptação Proativa na Web

Entende-se por adaptação *proativa* toda forma de adaptação produzida por algum mecanismo ou esquema, capaz de *antecipar* possíveis estados futuros do modelo e, se desejável, introduzi-los aos usuários. Assim a adaptação proativa busca induzir o surgimento de novos estados e configurações no modelo, segundo a linha seguida pelo usuário. A aplicação de técnicas de adaptação proativa pode conduzir a bons resultados (principalmente em escala de grupo), uma vez que acelera a convergência do modelo em direção a resultados (pontos ótimos locais) que de outra forma poderiam demorar mais tempo a ser atingidos, ou mesmo nem sequer ser considerados [PAL 98].

A arquitetura da web baseia-se no princípio da descentralização de operações e controle, o que possibilita uma interação muito mais direta entre produtores e consumidores de informações do que ocorre por exemplo na área de publicações, na televisão ou na indústria cinematográfica.

A organização *material* da web é produzida por seus projetistas que usam suas idéias intuitivas de estruturação do conhecimento e semântica na construção de nodos e sub-redes. Estas contribuições individuais são integradas gradualmente a corpos mais amplos de conhecimento preexistentes na web, assim expandindo o conhecimento da rede como um todo. Diversos elementos influenciam neste processo de organização de conhecimento, inclusive fatores psicológicos, lingüísticos, sociais e culturais, interesses e motivações dos autores e usuários das páginas web.

Esta organização uma vez materializada na rede assume uma representação dinâmica, objetiva e em geral de duração prolongada. Os novos nodos e links introduzidos modificam-se relativamente pouco ao longo do tempo. As modificações que ocorrem na rede são fruto de procedimentos de atualização, remoção e principalmente adição de conhecimento, sob a forma de nodos e links. Tais procedimentos produzem modificações concretas na rede e em geral são controlados pelos próprios produtores da informação. Eventuais inconsistências decorrentes de modificações materiais na rede, como links para nodos inexistentes ou nodos com features não suportadas por certos browsers, apresentam diversas formas de tratamento (nem sempre satisfatórias) como transferência de conexão, tentativa de localização do link por semelhança, etc.

Por outro lado, o que se poderia denominar organização *virtual* da rede é produzida pelo processo dinâmico de navegação, o qual decorre da percepção semântica associativa dos usuários, ativando determinados links, que lhes parecem mais promissores ou interessantes em detrimento de outros que o são menos. No atual estágio tecnológico da web o conhecimento produzido pelo processo de navegação não é registrado sistematicamente, sendo portanto desperdiçado. No entanto poderia ser empregado na construção de modelos auto-organizáveis de aprendizado associativo, em um sentido muito próximo ao proposto por Johan Bollen em [BOL 96] onde a web é vista como uma memória associativa de grandes dimensões.

O primeiro passo para tornar uma memória associativa mais eficiente, segundo diversas correntes da psicologia cognitiva [RUS 95], é permitir que ela própria descubra a melhor organização possível para si própria. Na mente humana, conhecimento e significado se desenvolvem mediante um processo de aprendizado associativo: os conceitos que são mais freqüentemente usados juntos se tornam mais fortemente conectados. É possível implementar mecanismos similares na web criando associações

com base nos caminhos percorridos pelos usuários através da rede de links. O princípio é simplesmente este: os caminhos percorridos por mais usuários se tornam *mais fortes*, enquanto que links raramente usados se tornam *mais fracos*.

Uma heurística simples pode então propor possíveis candidatos para novos links: Se um usuário navega de A para B e de B para C, é provável que exista não somente uma relação entre A e B mas também entre A e C (transitividade) e entre B e A (simetria), conforme apresentado na Tabela 4.1.

TABELA 4.1 - Operadores de fechamento

| OPERAÇÃO | EXPRESSÃO | OBSERVAÇÕES |
|----------------|--|--|
| FREQÜÊNCIA | $A \rightarrow B \rightarrow P_{AB} = P_{AB} + K_F$ | Se o link do nodo A para o nodo B é ativado, então seu potencial de ativação é incrementado de um valor constante K_F |
| TRANSITIVIDADE | $A \rightarrow B, B \rightarrow C \rightarrow P_{AC} = P_{AC} + K_T$ | Se a ativação ocorre de A para B e depois de B para C, o potencial de ativação entre A e C é incrementado pela constante K_T |
| SIMETRIA | $A \rightarrow B \rightarrow P_{BA} = P_{BA} + K_S$ | Se o link do nodo A para o nodo B é ativado, então o potencial de ativação de B para A é incrementado no valor de K_S |
| DEGRADAÇÃO | $\forall P, P = P - Z \mid T$ | A cada intervalo de tempo T, todo potencial de ativação P na rede é decrementado em um determinado percentual Z do seu valor corrente. |

Com a aplicação destes operadores sobre redes quantizadas é possível traçar a evolução da rede. Neste processo os operadores de frequência e degradação registram a *história* da rede, enquanto que os operadores de transitividade e simetria introduzem novas possibilidades de conexões. Desta forma, novos links potenciais seriam continuamente gerados (variedade) mas somente aqueles que obtivessem determinada *força* seriam retidos e tornados visíveis ao usuário (restrição), enquanto que aqueles que não atingissem tal *limiar de visibilidade* seriam mantidos ocultos. Um experimento realizado sobre um sistema hipertexto adaptativo [BOL 96] comprovou a exequibilidade desta proposta.

4.5 Parâmetros de Adaptação

O ajuste preciso dos valores dos parâmetros envolvidos na formulação dos operadores é de grande importância para a evolução apropriada do modelo. Tais parâmetros são:

- *Potencial de Ativação (P)*: O potencial de ativação de um link é a medida discreta da probabilidade de que o link seja ativado num próximo intervalo de tempo. Seu valor é obtido medindo-se a frequência com que cada link na

rede é ativado e aplicando sistematicamente os sobre esta os operadores de transitividade, simetria e degradação. O resultado deste processo é uma rede quantizada que representa proativamente o estado de evolução do sistema.

- *Limiar de Visibilidade (L)*: O limiar de visibilidade corresponde a um limite do potencial de ativação abaixo do qual os links não são mais visíveis. Seu valor pode ser graduado, para possibilitar a visão da rede em diferentes níveis limites, ou composto em um intervalo para observar diferentes camadas da rede.
- *Constante de Freqüência (K_F)*: Valor que é adicionado ao potencial de ativação de cada link toda vez que este é percorrido. Como as demais constantes, pode ser arbitrada. Supõe-se que seu valor deva ser superior ao das demais constantes, tendo em vista representar um evento concreto de navegação.
- *Constante de Transitividade (K_T)*: Valor que é sistematicamente adicionado ao potencial de ativação de um link segundo a regra de transitividade apresentada na tabela 4.1. Seu efeito sobre a rede deve ser menor do que a K_F , entretanto sua aplicação pode propagar-se e influir no potencial de ativação de diversos links na rede..
- *Constante de Simetria (K_S)*: Valor que é sistematicamente adicionado ao potencial de ativação de um link segundo a regra de simetria apresentada na tabela 4.1. Seu efeito é predominantemente local mas sua influência também se exerce sobre a rede como um todo.
- *Taxa de Degradação (Z)*: Valor percentual que decreta o potencial de ativação de um link a cada intervalo de tempo T considerado. O intervalo de tempo pode ser um minuto, uma hora ou um dia, conforme desejado.

A título de ilustração, uma possível configuração dos parâmetros de adaptação de um determinado sistema é dada na tabela 4.2.

TABELA 4.2 - Uma possível configuração dos parâmetros de adaptação

| Parâmetro | Valor |
|-----------------------------|--------------------|
| Limiar de Visibilidade | $L = 25$ |
| Constante de Freqüência | $K_F = 1,00$ |
| Constante de Transitividade | $K_T = 0,50$ |
| Constante de Simetria | $K_S = 0,20$ |
| Intervalo de Tempo | $T = 24 \text{ h}$ |
| Taxa de Degradação | $Z = 0.02$ |

Os parâmetros apresentados na tabela 4.2 caracterizam a representação proativa de uma rede quantizada onde $L = 25$ indica que somente serão apresentados ao usuário os links cujo potencial de ativação P seja igual ou superior a 25. A cada ativação o link recebe o incremento de uma unidade por conta da constante de frequência KF . No mesmo processo o link inverso recebe um incremento de 0,20 unidades em função da aplicação da regra de simetria. A aplicação do operador de transitividade é executada considerando os segmentos de rede que são grafos de duas setas com a conformação do diagrama (b) na figura 4.2. Isto significa que, ao contrário da frequência e da simetria, é necessário considerar *dois* passos de navegação para executar a regra da transitividade. Em um ciclo transitivo a rede é totalmente percorrida, em uma determinada ordem, e cada percurso completo representa um ciclo.

O mecanismo de degradação segue no exemplo o tempo físico, mas se poderia adotar alternativamente um tempo virtual (expresso em ciclos transitivos, por exemplo). Os valores $T = 24$ h e $Z = 0,02$, significam que a cada 24 h todo link na rede sofre um decremento de 2% em seu potencial de ativação. Aqui a opção por uma taxa ao invés de um valor constante garante que após a inicialização os links não vão *desaparecer* (potencial de ativação = 0). Com o uso de uma taxa, mesmo sem serem jamais ativados, os links continuam no sistema, (com um potencial de ativação indefinidamente pequeno), apesar de estarem ocultos do usuário por um limiar de visibilidade muito mais elevado.

A degradação concebida desta forma permite estabelecer conceitos como o de *meia vida* (V) de uma rede, calculada como o número de ciclos que um link qualquer na rede leva para degradar seu potencial de ativação à metade, no caso de não receber qualquer reforço por frequência, transitividade ou simetria. Na tabela 4.3 apresenta-se a meia vida da rede exemplificada para diferentes valores da taxa de degradação Z .

TABELA 4.3 - Meia vida de uma rede em função da sua taxa de degradação

| Z (a cada ciclo) | Meia vida (número de ciclos) |
|---------------------|---------------------------------|
| 0,01 | 69,97 |
| 0,02 | 35,31 |
| 0,03 | 23,76 |
| 0,04 | 17,98 |
| 0,05 | 14,51 |
| 0,10 | 7,58 |
| 0,15 | 5,27 |
| 0,20 | 4,11 |

A meia vida da rede é assim uma medida global que se aplica a todos os links e permanece constante enquanto permanecer constante a taxa de degradação da rede. O cálculo da meia vida é obtido pela aplicação direta de séries geométricas, segundo a fórmula:

$$V = 1 + \log(0.5) / \log(1 - Z)$$

4.6 O Processo de Adaptação

Como foi visto o processo de modelagem apresenta-se cíclico sob diversos pontos de vista. Uma visão algorítmica possível deste processo poderia apresentar os seguintes passos:

1. Inicializar a rede
2. Por um tempo indeterminado fazer
 - 2.1. Computar a frequência de ativação dos links da rede
 - 2.2. Aplicar os operadores de transitividade e simetria
 - 2.3. Aplicar o operador de degradação
 - 2.4. Salvar a rede

A inicialização da rede corresponde a delimitar um *domínio*, isto é, um segmento da rede sobre o qual se deseja efetuar o processo de modelagem e atribuir a cada link um potencial de ativação inicial. É necessário atentar para dois pontos aqui. Primeiro, o segmento escolhido para domínio pode ser tão grande quanto se deseje, condicionado entretanto na prática à capacidade computacional disponível. Em segundo lugar, o valor inicialmente atribuído ao potencial de ativação dos links da rede é aleatório, entretanto deve ser mantido relativamente baixo para evitar grandes distorções no início do processo. Essas considerações são de ordem prática e não tem impacto maior sobre a eficácia do modelo.

Uma questão interessante, relacionada à implementação do modelo é a possibilidade dos passos 2.1 a 2.4, que formam o laço de modelagem, serem executados de forma concorrente ou então, caso se torne desejável sua sincronização, num estilo *multithread* ou *pipeline*. Isto é possível, uma vez que estas ações são na verdade independentes entre si e podem ser executadas em paralelo em diferentes granulosidades.

Uma das principais vantagens do modelo de aprendizado associativo baseado em redes quantizadas reside no fato de que ele funciona localmente (é necessário armazenar informações sobre nodos que estão no máximo dois passos adiante), mas a organização que ele produz é global. Com tempo suficiente, nodos que se encontrassem a um número arbitrário de passos de distância poderiam se tornar diretamente conectados caso um número suficiente de usuários percorresse o caminho entre eles. É possível imaginar a extensão deste método com técnicas mais sofisticadas, que computassem, por exemplo o *tempo* que os usuários dedicam a cada nodo ou o grau de similaridade entre os nodos com base em seu conteúdo e empregar esta informação para sugerir nodos similares como links possíveis a partir de um determinado nodo [HEY 96]. O resultado esperado deste processo de aprendizado associativo é que os nodos que mais provavelmente serão utilizados em conjunto estarão também situados mais próximos na topologia da web.

Se tais algoritmos de aprendizado pudessem ser generalizados na web como um todo, o conhecimento ali existente poderia se estruturar em uma rede associativa em permanente aprendizado. Cada vez que um novo nodo fosse introduzido seus links iniciariam imediatamente a adaptar-se ao padrão de seu uso e novos links não previstos pelo autor poderiam nele surgir se o seu potencial ultrapassasse um determinado limiar. Uma vez que tal mecanismo de certa forma assimilaria o conhecimento coletivo de todos os usuários da web, supõe-se que os resultados obtidos viriam a ser muito mais úteis, abrangentes e confiáveis do que qualquer sistema de indexação concebido segundo os interesses de indivíduos ou grupos. A mesma idéia também se aplica a contextos onde a organização da rede se daria a partir de um determinado ponto de partida - um servidor ou uma rede de servidores em uma organização - espalhando-se sem limites por toda a web. A rede formada segundo tal principio certamente se aproximaria muito da semântica informal do grupo de usuários considerado, que poderia ser caracterizado sob muitos aspectos a partir desta representação.

O estudo das redes produzidas por auto-organização decorrente da observação da atividade de navegação na web pode levar ainda à identificação de estruturas e construções semânticas básicas, capazes de serem recombinaadas na formação de representações mais complexas com múltiplas finalidades, tais como a otimização de redes de informações, mapeamento e representação do conhecimento compartilhado por especialistas, o provimento de um padrão para a identificação de idéias, etc.

Capítulo 5

Um Modelo de Adaptação Simbólica

*Aspectos semânticos do processamento de informações são abordados no presente capítulo através da Teoria da Situação, estabelecendo um arcabouço formal para a representação, composição e inferência da relação de relevância entre os nodos de sistemas de Hipermídia Adaptativa. O ponto de partida são os conceitos de **ínfon**, **documento** e **descriptor**, assim como a semântica da relação “é sobre”, que pode existir entre diferentes documentos.*

5.1 Introdução à Teoria das Situações

O requisito essencial para o conceito nominal de informação é na realidade um processo de “informar”, isto é, um processo pelo qual *alguém* ou *alguma coisa* (um agente ou entidade) se torna *informado*. Esta idéia evoca a imagem da recepção de uma *mensagem* - uma peça de conhecimento - pela qual o agente receptor passa a deter a informação que é acomodada no seu particular *contexto mental* aumentando assim sua *quantidade* - e possivelmente a *qualidade* - de conhecimento. Isto pode ser visto também como o processo de solucionar um problema ou de selecionar uma resposta dentre uma coleção de respostas possíveis.

É usualmente aceito que um *incremento* de informação em um ponto do universo resulta num correspondente *decremento* de alguma outra coisa: da capacidade de armazenar informação futura, do número de problemas ainda por solucionar ou do número de possíveis estados do universo. Conforme foi apontado por Smithson [SMI 88], em geral apenas a *presença* da informação vinha sendo considerada pela teoria clássica da informação, enquanto sua *ausência* era vista simplesmente como *falta* de informação. Entende-se hoje que a presença e a ausência de informação (no sentido de incerteza, indeterminação) estão intimamente relacionadas e devem ser tratadas de forma integrada. A recepção de informação envolve necessariamente os elementos de *surpresa* e *novidade* - como uma mudança que poderia não ter sido prevista ou algo que era desconhecido até a recepção da informação. A ocorrência de novidade ou surpresa requer a existência de *liberdade* no sistema em que a novidade ocorre. Essa liberdade é então restrita pela *recepção* da informação.

Em última análise, segundo Ashby [ASH 56], os dois conceitos fundamentais de *variedade* e *restrição*, surgem aqui também como base da teoria da informação. Em outras palavras, um incremento na informação é um incremento na restrição, no “dar forma” ao universo com a correspondente redução em sua variedade. Naturalmente a informação também pode ser *perdida*, o que resulta no incremento da variedade. Os conceitos de variedade e restrição são muito gerais, contudo apresentam a vantagem de permitir um tratamento matemático de uma maneira clara e precisa.

Há entretanto uma grande controvérsia quanto ao uso do termo “*informação*”, principalmente quanto ao seu emprego na descrição de uma medida de *restrição*, conforme discutido por Shannon [SHA 56]. Em geral o termo tem sido empregado para designar o sentido *oposto*, por exemplo, na teoria clássica da comunicação de Shannon e

Weaver [SHC 64], emprega-se *informação* em associação com a quantidade de *variedade*, em termos da capacidade de um canal de comunicar informação.

Há também confusão na literatura sobre a distinção entre o conceito *sintático* de informação como uma medida de variedade ou restrição e os correspondentes conceitos *semânticos* que se relacionam com *significado*, *conteúdo* ou *valor* e outras noções semelhantes da semiótica e da lingüística. Enquanto que o primeiro é de grande exatidão, os demais podem apresentar muita subjetividade. Muitos pesquisadores, por essa razão, têm evitado a questão deixando de usar o termo *informação* e substituindo-o por *incerteza* ou outras expressões mais específicas.

O conceito de indeterminismo e sua relação com a idéia de incerteza é também importante. *Indeterminismo* é usualmente assumido como sendo uma propriedade *ontológica* de algum processo em um sistema *físico*, enquanto que *incerteza* é uma propriedade *epistêmica* de algum sistema de *modelagem*. Assim, a presença de indeterminismo em um sistema objeto irá acarretar a presença de incerteza em um modelo apropriado desse sistema e a incerteza no modelo deverá refletir o indeterminismo presente no sistema objeto. É preciso notar entretanto que a presença de incerteza em um modelo não acarreta necessariamente o correspondente indeterminismo no sistema objeto. Uma vez que o conhecimento acumulado por um sistema é tão bom quanto os modelos que ele consegue construir, a presença de indeterminismo como propriedade ontológica dos sistemas reais passa a ser então uma questão metafísica do processo de modelagem.

As relações existentes entre diferentes estruturas de informação constituem o ponto focal do presente capítulo, no qual se adota uma abordagem onde estruturas mais complexas de informação denominadas *situações* são produzidas a partir da ação de operadores sobre elementos primitivos denominados *ínfons*. Esses operadores têm sua ação baseada em diferentes aspectos da semântica do relacionamento “*é sobre*” que pode existir entre duas ou mais peças de informação.

5.2 Relevância

Pode-se pensar em *recuperação de informações* como a materialização do problema de acessar, com o processamento necessário, a informação que é provavelmente *relevante* para uma certa necessidade. A noção de que uma determinada peça de informação é “*sobre*” alguma coisa está intimamente associada com a *relevância* que a informação apresenta em determinada *situação*. Se uma peça de informação é *sobre* uma necessidade manifesta de informação (usualmente uma *consulta*) então esta peça é considerada relevante com respeito à informação solicitada. Analisa-se aqui os conceitos relacionados com a qualificação da *relevância situada* de uma determinada peça de informação a partir da teoria geral da informação [HUI 94] [BRP 94]. Esses conceitos surgem da *Teoria da Situação* [BAR 89] [BAR 90] [DEV 91] que abrange uma teoria da informação suficientemente poderosa para estudar a relação entre certa informação e aquilo *sobre* o que ela é.

Definição 5.1: Ínfon

Um *ínfon* é uma estrutura $\langle\langle R, a_1, \dots, a_n; i \rangle\rangle$ que representa a informação de que a relação *R* vale (se $i=1$) ou não vale (se $i=0$) entre os objetos a_1, \dots, a_n . ■

Os objetos desta definição referem-se a peças de informação atômicas, como palavras-chave, sentenças, imagens, sons, etc. O conjunto de relacionamentos pode por

exemplo consistir em relações semânticas [HAD 95], relações conceituais [CHE 92], relações lingüísticas [BRU 93] ou relações informais [LAL 93]. O valor i é denominado a *polaridade* do ífon. Se a polaridade for 1 o ífon é dito *positivo* e *negativo* em caso contrário.

Em geral a informação contida em documentos é parcialmente modelada por um conjunto de termos denominados palavras-chave. O uso de palavras-chave na modelagem de informação resulta em ífons muito primitivos, que podem mesmo ser considerados como partículas sub-informacionais que estão para a informação assim como os prótons estão para os átomos. Para designá-los, Huiber e Bruza introduziram em [HUI 94] o termo *prófon*. Prófons são uma subclasse dos ífons baseada em uma relação I que denota uma relação unária não-específica refletindo a informação de si própria. Por exemplo se um documento D é indexado pela palavra chave “Alice” pode-se considerar que a informação “Alice” é *inerente a* ou *vale* em D . A notação

$$\langle\langle I, Alice ; 1 \rangle\rangle$$

é empregada para denotar o correspondente prófon. Para abreviar, prófons positivos desse tipo serão denotados aqui simplesmente como $\langle\langle Alice \rangle\rangle$.

5.3 Combinação de Informações

Pretende-se que os ífons capturem informação básica sobre objetos mais elaborados como documentos. Uma característica desta informação básica é que ela pode ser manipulada no sentido de formar estruturas de informações mais complexas. Por exemplo, duas peças de informação podem ser combinadas para formar uma terceira. Ao nível dos ífons esta combinação é obtida por meio de operadores especiais como *fusão*, *combinação* e *degradação* [DEV 91], que se passa a comentar:

5.3.1 Fusão

Tome-se por exemplo as palavras-chave “Alice” e “Aventuras”. Estas palavras podem ser combinadas para formar a frase “Aventuras de Alice”. Isto é um exemplo de *fusão informacional*, onde os itens de informação original são firmemente acoplados em uma nova estrutura com identidade própria. “Aventuras de Alice” traduz precisamente a informação surgida da fusão de “Alice” com “Aventuras”.

A fusão é modelada ao nível dos ífons pelo operador \oplus : $\langle\langle Alice \rangle\rangle \oplus \langle\langle Aventuras \rangle\rangle$. O resultado é um ífon $\langle\langle Possui, \langle\langle Alice \rangle\rangle, \langle\langle Aventuras \rangle\rangle ; 1 \rangle\rangle$, que se baseia em duas palavras-chave combinadas pela relação “Possui” retirada de um conjunto de relações básicas pré-definidas.

Naturalmente é possível fundir ífons de diferente aridade que se relacionam entre si. Por exemplo a fusão do ífon $\langle\langle Possui, \langle\langle Alice \rangle\rangle, \langle\langle Aventuras \rangle\rangle ; 1 \rangle\rangle$ com o prófon $\langle\langle País das Maravilhas \rangle\rangle$ resulta, através da relação “Local” no ífon $\langle\langle Local \langle\langle Possui, \langle\langle Alice \rangle\rangle, \langle\langle Aventuras \rangle\rangle ; 1 \rangle\rangle, \langle\langle País das Maravilhas \rangle\rangle ; 1 \rangle\rangle$. Deve-se notar que para cada elemento do conjunto de relações existe um único operador com as suas próprias propriedades. Assim se rotula o operador com a relação r - e se escreve \oplus_r - sendo r tomada do conjunto de relações básicas considerado.

5.3.2 Composição

A fusão combina peças de informação através do emprego de um relacionamento que as “funde” em uma nova peça de informação. Algumas vezes pode ser de interesse combinar certos itens de modo “frouxo” ou representar um acoplamento fraco em que os componentes se mantêm relacionados, porém sem que a semântica da relação seja essencial. Por exemplo, na interpretação de $i \wedge j$, segundo a teoria dos modelos, não é necessário que i e j sejam relacionados semanticamente. Esta intuição conduz ao que se denomina *composição de informação*. Seja por exemplo a expressão $\langle\langle \text{Alice} \rangle\rangle \wedge \langle\langle \text{Rainha de Copas} \rangle\rangle$. Intuitivamente o infon expressa que há informação sobre $\langle\langle \text{Alice} \rangle\rangle$ e sobre $\langle\langle \text{Rainha de Copas} \rangle\rangle$ em uma determinada situação. Entretanto tal informação pode não estar relacionada como por exemplo em “*Alice conversava com o Coelho Branco enquanto o Chapeleiro Louco visitava a Rainha de Copas*”.

5.3.3 Degradação

A operação de *degradação da informação* é assim denominada porque acontece quando a informação contida em um infon deixa de ser uma certeza. Assim, em determinadas situações, é necessário degradar um infon composto. Na recuperação de informações por exemplo, infons compostos são empregados normalmente para caracterizar a *necessidade* de informação e não é necessário que esta esteja reunida em uma única peça ou *cluster*. A degradação de informação envolve incerteza. Por exemplo, analisando a consulta:

Encontrar um livro sobre “lógica de primeira ordem” ou “teoria da situação”

Seu conteúdo pode ser expresso por um infon composto:

$\langle\langle \text{lógica de primeira ordem} \rangle\rangle \vee \langle\langle \text{teoria da situação} \rangle\rangle$

Este tipo de incerteza, por razões práticas não é empregado para fins de indexação. Entretanto pode vir a ser muito útil, por exemplo, na presença de ambigüidades. Pode-se descrever um infon através de dois outros compostos pelo operador de degradação. A palavra-chave “*banco*”, por exemplo, pode se referir a um “*banco de praça*” ou a um “*banco financeiro*”. Assim ao invés de empregar o prófon $\langle\langle \text{banco} \rangle\rangle$ se poderia utilizar:

$\langle\langle \text{banco de praça} \rangle\rangle \vee \langle\langle \text{banco financeiro} \rangle\rangle$

5.3.4 Contenção

Um conceito de grande importância para o estudo aqui desenvolvido é o de *inferência de informação*. A idéia corresponde a inferir informação adicional que não esteja explícita nos itens de informação disponíveis. Em sistemas de recuperação de informação a inferência baseia-se na noção de *contenção*. Por exemplo, a partir do infon $\langle\langle \text{Alice} \rangle\rangle \wedge \langle\langle \text{Rainha de Copas} \rangle\rangle$, pode-se inferir o prófon $\langle\langle \text{Alice} \rangle\rangle$ uma vez que este último está contido no primeiro. De acordo com Barwise e Etchemendy [BAR 90] os infons podem ser parcialmente ordenados com respeito à contenção de informação (denotada por \rightarrow).

Definição 5.2:

$i \rightarrow j$ se e somente se o ífon j pode ser derivado do ífon i no contexto do modelo de recuperação considerado. ■

Por exemplo, em modelos booleanos uma derivação de j a partir de i é possível se $i \vdash j$. Ao nível de controle em sistemas de recuperação de informações esta derivação baseia-se na relação subconjunto. Em teoria da informação esta relação é assumida como reflexiva, anti-simétrica e transitiva. A propriedade de contenção para as operações de composição e degradação pode ser diretamente especificada:

$$i \wedge j \rightarrow i \rightarrow i \vee j$$

Em relação à operação de fusão a propriedade $i \oplus, j \rightarrow i \wedge j$ é normalmente válida, entretanto isso irá depender da especificação da relação r .

5.3.5 Preclusão

Nem todos os ífons podem ser combinados de forma significativa. Isto acontece em virtude da possibilidade de ocorrência de informação contraditória no sistema. Diz-se que os ífons i e j encontram-se em preclusão um com o outro e isto é denotado por $i \perp j$. É natural assumir por exemplo que um ífon com polaridade 1 preclui o mesmo ífon com polaridade 0. Por exemplo

$$\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle \perp \langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 0 \rangle\rangle$$

A noção de preclusão é considerada fundamental para uma teoria da informação [LAN 86]. Um de seus principais empregos é no detalhamento da informação. Por exemplo, se uma peça de informação for caracterizada pelo ífon $\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle$ é possível assumir a hipótese de que este ífon encontra-se em preclusão com $\langle\langle \text{Lutando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle$ (partindo do princípio que enquanto alguém fala não há luta). Pode-se então assumir (por exemplo, por default) que a peça de informação considerada também contém (implicitamente) a informação de que $\langle\langle \text{Lutando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 0 \rangle\rangle$. Portanto pode ser relevante para a pesquisa da informação “*Alice não está lutando com o Gato*”.

5.3.6 Ífons Parametrizados

O conceito de *parâmetros* em ífons na teoria da situação é também de grande utilidade na teoria da informação. Por exemplo, para deixar claro no exemplo anterior que “*Alice*” é uma PESSOA, ou que a relação *falando* se dá entre uma PESSOA e um ANIMAL. Emprega-se aqui letras maiúsculas (A, B , etc.) para denotar tais parâmetros. Por exemplo, o ífon parametrizado:

$$\langle\langle \text{Falando}, A, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle \text{ com } A=OBJ | \langle\langle \text{Pessoa}, OBJ; 1 \rangle\rangle$$

descreve intuitivamente que uma pessoa A está falando com um “*Gato*”. O ífon composto

$$\langle\langle \text{Falando}, A, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle \wedge \langle\langle \text{Nome}, \langle\langle \text{Alice} \rangle\rangle, A; 1 \rangle\rangle$$

estabelece o fato de que a pessoa que está falando com o “*Gato*” tem o nome de “*Alice*” (porque ambos os ífons apresentam o mesmo parâmetro A).

Até aqui se comentou o uso de ínfons e prófons como um mecanismo para a representação de peças de informação. Resta entretanto analisar como documentos e consultas devem ser modelados. Um documento *suporta* uma peça de informação e como tal pode ser modelado como um conjunto de ínfons. Estes conjuntos de ínfons são denominados *situações* e são representações abstratas da informação contida no documento considerado. Devlin [DEV 91] denominou esse tipo de situação de *situação abstrata*, uma vez que são construídas matematicamente a partir de um conjunto de ínfons. Assim as situações reais (que fazem parte do mundo real) podem ser caracterizadas com ínfons para formar situações abstratas. Este processo denomina-se *digitalização da informação* [DRE 81].

5.4 Ínfons, Documentos e Situações

Os ínfons podem ser vistos como partículas elementares de informação descrevendo um documento, com \models como relação de *suporte*. Isto pode ser ilustrado da seguinte maneira: Seja D um documento contendo o texto do livro “*Aventuras de Alice no País das Maravilhas*”. Na representação aqui proposta, diz-se que este documento suporta, entre outros o ínton:

$$D \models \langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle$$

Este ínton expressa que um documento D suporta a informação de que “Alice está falando com um Gato”. Em termos de recuperação de informações isto significa que este fato está presente no documento D. Assume-se que a relação de suporte é fechada sob a contenção de informações.

Definição 5.3

Se um documento D suporta um ínton i, diz-se que i é um descritor de D. $d(D)$ é o conjunto dos descritores do documento D. Vale a relação $D \models i \Leftrightarrow i \in d(D)$. ■

Pode acontecer que um certo ínton não seja suportado por um determinado documento. A intuição por trás disso é que a informação apresentada pelo ínton não é inerente ao documento considerado. Por exemplo, para o mesmo documento D:

$$D \not\models \langle\langle \text{Falando}, \langle\langle \text{Romeu} \rangle\rangle, \langle\langle \text{Julieta} \rangle\rangle; 1 \rangle\rangle$$

Deve-se notar que a expressão acima é muito diferente de uma situação suportar a negação de um fato. Isto pode ser realizado por ínfons com polaridade 0. Por exemplo a relação:

$$D \models \langle\langle \text{Concorda}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 0 \rangle\rangle$$

expressa que o documento D *suporta* o fato de que *Alice* e o *Gato* não concordam.

5.4.1 Tipos

Na teoria da situação também há uma generalização de ínfons para tipos. Sejam por exemplo os três ínfons a seguir:

$$\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle$$

$$\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Rainha de Copas} \rangle\rangle; 1 \rangle\rangle$$

$$\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Coelho Branco} \rangle\rangle; 1 \rangle\rangle$$

Estes três ínfons estão descrevendo a mesma situação: “*Alice está falando*”. A única diferença entre eles é com quem Alice está falando. Para os três ínfons acima há um tipo unificador que pode ser representado por:

$$\varphi = [S \mid S \models \langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, P; 1 \rangle\rangle]$$

com P representando um objeto com o qual se pode falar. Este tipo denota todos os documentos em que *Alice está falando*. Além disso, todo o documento que suporta esta informação é dito ser do *tipo* φ (denota-se $D : \varphi$).

5.5 Combinação de Situações

Os ínfons constituem o nível mais baixo da granulosidade da informação. Em um nível mais alto se encontram os *documentos*, *situações* e *consultas*, na terminologia da recuperação de informações. Haveria uma maneira natural de generalizar os operadores do nível dos ínfons para o nível dos documentos? Uma primeira hipótese seria trabalhar diretamente com o conteúdo dos documentos, o que é muito difícil dada a extensão do espaço inferencial. Outra possibilidade, que se irá perseguir aqui, seria associar a cada documento D um conjunto $d(D)$ de ínfons descrevendo D, denominado o *descriptor* de D. Os descriptors do tipo $d(D)$ pertencem a uma classe especial de situação, cujos ínfons descrevem um objeto externo. Como são em geral muito menores que o documento que descrevem e podem reduzir a descrição a um pequeno número de ínfons preestabelecidos, como título, palavras-chave, etc., os descriptors são muito mais facilmente operados por um sistema de inferência. Assim as situações representadas pelos descriptors podem ser combinadas de três modos distintos na formação de uma terceira: por *união*, por *fusão* ou por *composição*, como se descreve a seguir.

5.5.1 União

A união de duas situações S e T corresponde a tomar todos os ínfons de S juntamente com todos os ínfons de T. Isto é denotado por $S \cup T$ e representa todos os ínfons das duas situações originais. Assim se a situação $S \cup T$ suporta um ínfon, isto significa que a situação S ou a situação T suporta esse ínfon. O operador de união de situações pode ser usado para modelar o fato de que se adicionou informação a uma dada situação. Os parâmetros empregados no primeiro conjunto de ínfons são assumidos diferentes dos parâmetros empregados no segundo. Assim $\{\langle\langle \text{Nome}, \langle\langle \text{Alice} \rangle\rangle, P; 1 \rangle\rangle\}$ unido com $\{\langle\langle \text{Nome}, \langle\langle \text{Alice} \rangle\rangle, P; 1 \rangle\rangle\}$ irá resultar no conjunto $\{\langle\langle \text{Nome}, \langle\langle \text{Alice} \rangle\rangle, P1; 1 \rangle\rangle, \langle\langle \text{Nome}, \langle\langle \text{Alice} \rangle\rangle, P2; 1 \rangle\rangle\}$ que terá *dois* parâmetros. Isto permite modelar o fato de que duas pessoas com o nome *Alice* podem não ser a mesma.

5.5.2 Fusão

O operador de fusão de situações, denotado por \otimes pode ser visto como a criação de novas situações baseadas em outras situações que são postas em relação uma com a outra. A título de ilustração, considerem-se duas situações, uma na qual “*Alice fala*” e outra em que “*alguém fala com o Gato*”. É possível criar nova informação a partir delas estabelecendo-se que “*Alice fala com o Gato*”. Isto é baseado na hipótese de que há uma certa relação (Alice é “alguém”) entre as duas situações.

Há diferentes maneiras de definir este tipo de fusão entre duas situações. Uma delas ocorre através da composição de cada ífon na primeira situação com todos os ífons da segunda. Uma outra definição pode ser feita considerando apenas a combinação de ífons particulares. Por exemplo, sejam os conjuntos $S = \{\langle\langle Faland\grave{o}, P, Q; 1 \rangle\rangle, \langle\langle Nome, \langle\langle Alice \rangle\rangle, P; 1 \rangle\rangle\}$ e $T = \{\langle\langle Faland\grave{o}, U, V; 1 \rangle\rangle, \langle\langle Nome, \langle\langle Gato \rangle\rangle, V; 1 \rangle\rangle\}$. Para deixar claro que “*Alice esta falando com o Gato*” deve-se estabelecer que na uniao de S e T os parametros P e U sao os mesmos, assim como respectivamente Q e V. O resultado sera o conjunto $S \otimes T$ cuja expressao e:

$$\{\langle\langle Faland\grave{o}, U, V; 1 \rangle\rangle, \langle\langle Nome, \langle\langle Alice \rangle\rangle, U; 1 \rangle\rangle, \langle\langle Nome, \langle\langle Gato \rangle\rangle, V; 1 \rangle\rangle\}$$

Este tipo de substituiao de parametros e definido da seguinte maneira:

Definiao 5.4: Substituiao de Parametros

A notao $S^{(X,Y)}$ representa a substituiao no conjunto S do parametro X pelo parametro Y. As propriedades de substituiao de parametros sao definidas como se segue:

$$S^{(X,Y,Z)} = (S^{(X,Y)})^{(Y,Z)}$$

$$S^{(W,X)}(Y,Z) = (S^{(W,X)})^{(Y,Z)} \quad \blacksquare$$

Assim a definiao de fusao com substituiao de parametros pode ser dada por

$$S \otimes T = (S \cup T)^{(A_1, \dots, A_n), (Z_1, \dots, Z_n)}$$

No exemplo anterior, “*Alice esta falando com o Gato*” pode entao ser formalizado como $S \otimes T = (S \cup T)^{(P,U)(Q,V)}$. De modo semelhante a fusao de ífons, o resultado de uma fusao de situaoes depende da forma como e definida e diversos operadores de fusao podem ser propostos. Assim se emprega o smbolo \otimes_x rotulado com a particular relaao de fusao empregada (no caso, x).

5.5.3 Composiao

O resultado da composiao de situaoes, denotado por $S \cap T$, e uma nova situaao que captura todos os fatos comuns as duas situaoes originais. Em recuperaao de informaoes este operador e util, por exemplo, na clusterizaao de documentos, permitindo gerar um sub-documento (a situaao composta), representando a situaao que esta simultaneamente disponivel nos dois documentos. Se este sub-documento for grande, e razoavel concluir que e mais provavel que os dois documentos sejam sobre assuntos semelhantes do que se o sub-documento for pequeno.

Na tabela 5.1, a seguir, apresenta-se um breve resumo dos operadores sobre ínfons e sobre situações até aqui comentados.

TABELA 5.1 - Operações sobre Ínfons e Situações

| OPERAÇÃO | EXPRESSÃO | OBSERVAÇÕES |
|-------------------------|-----------------------|--|
| Fusão de Ínfons | $i \oplus_r j \in S$ | depende da semântica de r |
| Composição de Ínfons | $i \wedge j \in S$ | se e somente se ($i \in S$ e $j \in S$) |
| Degradação de Ínfons | $i \vee j \in S$ | se e somente se ($i \in S$ ou $j \in S$) |
| União de Situações | $i \in S \cup T$ | se e somente se ($i \in S$ ou $i \in T$) |
| Composição de Situações | $i \in S \cap T$ | se e somente se ($i \in S$ e $i \in T$) |
| Fusão de Situações | $i \in S \otimes_x T$ | depende da definição de x |

5.5.4 A Relação “é sobre” entre Documentos

Encontram-se na literatura diversas abordagens à relação “é sobre”. Por exemplo como “*relacionamento entre tópicos*” [COO 71], “*sobre*” [MAR 77], “*possibilidade de conter informações sobre*” [RIJ 93] e “*correspondente a*” [NIE 90]. Formalmente representa-se a relação de “ser sobre” com o símbolo \downarrow . Assim, $D \downarrow D'$ significa intuitivamente que o documento D é sobre o documento D' . Correspondentemente, $D \not\downarrow D'$ será empregado para denotar que D não é sobre D' . Segundo Huibert e outros [HUI 95] [BAR 90] pode-se considerar a relação de “ser sobre” como um conjunto de derivações *mais ou menos* lógicas, que desempenham um papel de grande importância tanto na recuperação de informações quanto na teoria da situação. Por exemplo, dado que $S \cup T$ é sobre S , pode-se derivar que a situação $\{\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle, \langle\langle \text{Ouvindo}, \langle\langle \text{Gato} \rangle\rangle, \langle\langle \text{Alice} \rangle\rangle; 1 \rangle\rangle\}$ é sobre a situação $\{\langle\langle \text{Falando}, \langle\langle \text{Alice} \rangle\rangle, \langle\langle \text{Gato} \rangle\rangle; 1 \rangle\rangle\}$. Como se pode ver, na maioria dos casos a relação *é sobre* pode ser descrita em termos de axiomas e regras. Isto ocorre naturalmente por exemplo na derivação com o cálculo proposicional e em diversas lógicas modais [HUG 84]. A partir daqui a construção de um contexto para uma *teoria da informação situada* principia pela definição de uma linguagem L necessária para descrever o sistema de derivação.

Definição 5.5: A Linguagem L .

A linguagem L é definida como uma quádrupla $\langle I, S, Sb, F \rangle$, onde cada um dos componentes é descrito como se segue:

I é o conjunto dos *ínfons*, definido por:

- Se $i, j \in I$, então $i \wedge j$, $i \vee j$, $i \oplus_r j \in I$, $\forall r \in R$ (conjunto de relações sobre os objetos de L).

S é o conjunto das *situações*, definido por:

- Se $i \in I$ então $\{i\} \in S$;
- Se $A, B \in S$ então $A \cup B, A \cap B, A \otimes_x B \in S$ (para qualquer definição de fusão x).

Sb é o conjunto de fórmulas da relação “é sobre”, definido por:

- Se $A, B \in S$ então $A \downarrow B, A \downarrow B \in Sb$.

F é o conjunto de todas as **fórmulas** possíveis, definido por

- Se $p \in Sb$ então $p \in F$;
- Se $i, j \in I$ então $i \rightarrow j, i \perp j \in F$ ■

A definição de um sistema de derivação para a linguagem L foi introduzida em [HUI 95] nos seguintes termos:

Definição 5.6: Sistema de Derivação

Um sistema de derivação A é um par da forma $\langle Ax, Reg \rangle$ onde Ax é um conjunto de axiomas e Reg é um conjunto de regras da forma $R(T_1, \dots, T_k, T_{k+1})$, onde T_1, \dots, T_k são as premissas da regra R e T_{k+1} é a conclusão. ■

Assume-se que os axiomas, premissas e conclusões são elementos de F . O principal interesse aqui são as conclusões que pertencem ao conjunto Sb , isto é, que estabelecem um relacionamento *sobre*. Note-se que não se especificou como a relação de derivação é determinada pelo sistema de derivação. É possível adotar o mecanismo usual da lógica clássica: uma relação *é sobre* entre duas situações pode ser derivada se há entre elas uma seqüência de passos que ou são axiomas ou decorrem de passos anteriores pela aplicação de alguma regra. É possível, entretanto, adotar outras relações de derivação, por exemplo, para o tratamento de incerteza ou para a descrição de fenômenos modais, dinâmicos, etc. Assim o sistema de derivação aqui definido constitui um arcabouço suficientemente abstrato no qual é possível implementar diversos mecanismos de inferência preservando a unidade conceitual original.

5.6 Raciocínio sobre Situações

A relação “é sobre” é tratada aqui como uma relação entre *situações*, representando uma noção fundamental para as informações nelas contidas. Esta abordagem difere de outras, em que a relação “é sobre” pode ser expressa em termos de contenção de informações. Na presente seção é apresentado um conjunto de axiomas e regras que estabelecem algumas propriedades da relação “é sobre” ocorrendo entre situações. Os axiomas não devem ser considerados em um sentido estritamente lógico. Alguns deles não são universalmente válidos e somente valerão no contexto de sistemas particulares.

5.6.1 Postulados Básicos

Os três postulados básicos do raciocínio sobre situações são *reflexividade*, *transitividade* e *simetria*. Sua definição é bastante intuitiva, como pode ser observado na tabela 5.2.

TABELA 5.2 - Postulados Básicos do Raciocínio sobre Situações

| POSTULADO | EXPRESSÃO | OBSERVAÇÕES |
|----------------|--|--|
| Reflexividade | $S \downarrow S$ | Uma situação é sobre si própria. |
| Transitividade | $\frac{S \downarrow T \quad T \downarrow U}{S \downarrow U}$ | Se S é sobre T e T é sobre U , então S é sobre U |
| Simetria | $\frac{S \downarrow T}{T \downarrow S}$ | Se S é sobre T então T é sobre S . |

O axioma da *reflexividade* é trivial e indica simplesmente que uma situação é sobre si própria. A regra da *transitividade* é de grande importância em sistemas de recuperação de informações e estabelece que se S é sobre T e T é sobre U , então S é sobre U . Entretanto a transitividade não é válida em sistemas onde possa haver sobreposição. O fato de haver uma sobreposição entre S e T e outra entre T e U não implica em que haja sobreposição entre S e U .

A regra da *simetria* estabelece que não há diferença em concluir que uma situação S é sobre uma situação T ou vice-versa. A principal finalidade da simetria é promover recorrência. Ao nível de derivação de situações a simetria pode ser vista também como a derivação de uma inferência plausível.

5.6.2 Postulados de Combinação

Os postulados de combinação de situações são apresentados na Tabela 5.3.

A união de situações \cup e a composição \cap , devido às propriedades herdadas da teoria dos conjuntos são naturalmente *comutativas*. Em situações de fusão \otimes_x isto não é pré-definido e deve-se especificar, para cada definição de fusão x , se ela será comutativa ou não com o emprego do axioma apresentado.

A *união monotônica à esquerda* é uma regra de simples entendimento no relacionamento entre situações. O termo “*monotônica*” significa que a relação “é sobre” é preservada sob a união da situação colocada à esquerda do símbolo \downarrow da relação original. Por exemplo, dado que o conjunto $\{\langle\langle Alice \rangle\rangle, \langle\langle Gato \rangle\rangle\}$ é sobre $\{\langle\langle humano \rangle\rangle, \langle\langle gato \rangle\rangle\}$ uma nova situação é formada pela unificação informacional do primeiro conjunto com $\{\langle\langle Falando, \langle\langle Alice \rangle\rangle, \langle\langle Gato \rangle\rangle; 1 \rangle\rangle\}$. A união monotônica à esquerda permite concluir que este novo conjunto unido é também sobre $\{\langle\langle humano \rangle\rangle, \langle\langle gato \rangle\rangle\}$. A adição de informação conduz apenas a mais conclusões e nunca a uma redução destas. No caso da *união monotônica à esquerda forte*, tem-se a evidência de que o conjunto U que será unido com S também é sobre T . Esta regra permite estender S somente com informação já reconhecida como sendo sobre T .

A propriedade de *composição* expressa que se uma situação S é sobre uma dada situação T , esta relação é preservada sob a composição da situação T . No contexto da recuperação de informações pode-se pensar em reduzir um consulta (por exemplo, da forma “poluição do ar” para a forma “poluição”) o que sempre irá conduzir a um número maior (ou pelo menos igual) de documentos relevantes. A *união livre de contexto* estabelece que se S é sobre T e S também é sobre U então é possível unir a informação de T e U e concluir que S é sobre esta união. A recuperação booleana, por exemplo, é fundamentada neste princípio. Assim, se um documento D é sobre $\{\langle\langle Alice \rangle\rangle\}$ e o mesmo documento é sobre $\{\langle\langle Gato \rangle\rangle\}$ é possível assumir que o documento D é sobre $\{\langle\langle Alice \rangle\rangle, \langle\langle Gato \rangle\rangle\}$.

Nela representa-se o descritor simplificado de um documento identificado por A . O documento intitula-se *LogicWeb*, foi escrito por *Seng Wai Loke*, a última atualização foi efetuada em 7 de setembro de 1998 e o documento possui as palavras-chave *programação em lógica*, *prolog* e *world wide web*. O valor de G está fixado em 0.82 para com o documento B , e 0.17 para com o documento C . Isto indica que B é muito relevante para o conteúdo de A , enquanto que C somente um pouco.

TABELA 5.3 - Postulados de Combinação

| POSTULADO | EXPRESSÃO | OBSERVAÇÕES |
|-----------------------------------|--|--|
| Comutatividade | $S \otimes_x T \downarrow T \otimes_x S$ | Deve ser explicitada em cada descrição de fusão x . |
| União Monotônica à Esquerda | $\frac{S \downarrow T}{S \cup U \downarrow T}$ | Adicionar informação à situação S preserva a relação com T . |
| União Monotônica à Esquerda Forte | $\frac{S \downarrow T \quad U \downarrow T}{S \cup U \downarrow T}$ | Estende S com mais informação sobre a situação T . |
| Composição | $\frac{S \downarrow T}{S \downarrow T \cap U}$ | Se S é sobre T , isto é preservado pela composição de T . |
| União Livre de Contexto | $\frac{S \downarrow T \quad S \downarrow U}{S \downarrow T \cup U}$ | Se S é sobre T e sobre U , também é sobre a união de T com U . |
| Fusão Monotônica à Esquerda | $\frac{S \downarrow T}{S \otimes_x U \downarrow T}$ | Se S é sobre T , isto é preservado pela fusão de S , dependendo de x . |
| Fusão Monotônica à Esquerda Forte | $\frac{S \downarrow T \quad U \downarrow T}{S \otimes_x U \downarrow T}$ | Se S e U são sobre T a sua fusão também o é, dependendo de x . |
| Propriedades de Fusão | $S \otimes_x T \downarrow S \otimes_y T$ | Permite mapear entre duas definições de fusão (aqui x e y). |

Para cada definição de operador de fusão é necessário definir suas propriedades por meio do relacionamento x associado. As regras de fusão na tabela 3 poderiam ser adotadas, dependendo do contexto da representação. Assim formulou-se a *fusão monotônica à esquerda* e sua variante mais cautelosa, a *fusão monotônica à esquerda*

forte, de modo similar às correspondentes operações de união anteriormente comentadas. Esta última pode ser empregada para modelar a fusão em um nível mais baixo de granularidade da informação (por exemplo ao nível de considerações lingüísticas). Para o mesmo propósito pode-se pensar em mapeamentos entre as propriedades de diferentes relações de fusão.

5.6.3 Postulados Baseados em Ínfons

Os postulados apresentados até aqui não levam em conta as propriedades dos ínfons. Por exemplo, a relação de contenção \rightarrow é definida como uma regra de envolvimento de informação e pode ser útil em sistemas de recuperação de informações. Considera-se em geral o emprego de três postulados baseados em ínfons, apresentados na Tabela 5.4.

A regra de *união-contenção* estabelece que todas as regras válidas para a inferência de informações são válidas para a decisão da existência do relacionamento “é sobre”. Se há por exemplo um conjunto S que contém um ínfon i sobre um conjunto T que também contém i e é dado que $i \rightarrow j$, então pode-se substituir o ínfon i pelo ínfon j em T preservando a relação “é sobre” entre S e T . Assim a relação de contenção \rightarrow é transitiva, reflexiva e anti-simétrica (um reticulado distributivo) e portanto a decisão do relacionamento “é sobre” também possuirá estas três propriedades, configurando uma ordenação parcial nos sistemas de ínfons e de situações.

A *fusão-contenção* estabelece que se uma situação S , relacionada com uma situação $\{i\}$, é sobre uma situação T relacionada com a mesma situação $\{i\}$, então, dado que $i \rightarrow j$ pode-se concluir que a situação S relacionada com i é sobre a situação T relacionada com j . Finalmente o uso da relação de *preclusão* entre os ínfons pode ser proposto como uma versão restrita da união monotônica. Por exemplo, pode-se estabelecer que uma situação S é sobre uma situação T e então a adição de um ínfon i a S e um ínfon j a T irá preservar esta relação desde que não haja preclusão entre i e j .

TABELA 5.4 - Postulados Baseados em Ínfons

| POSTULADO | EXPRESSÃO | OBSERVAÇÕES |
|------------------------------|---|---|
| União-Contenção | $\frac{i \rightarrow j \quad S \cup \{i\} \downarrow T \cup \{i\}}{S \cup \{i\} \downarrow T \cup \{j\}}$ | O relacionamento “é sobre” é preservado pela união-contenção. |
| Fusão-Contenção | $\frac{i \rightarrow j \quad S \otimes_x \{i\} \downarrow T \otimes_x \{i\}}{S \otimes_x \{i\} \downarrow T \otimes_x \{j\}}$ | Idem para a fusão-contenção |
| Monotonicidade Composicional | $\frac{S \downarrow T \quad i \perp j}{S \cup \{i\} \downarrow T \cup \{j\}}$ | Se i não preclui j a composição é monotônica. |

5.6.4 Negação

Para o estudo do relacionamento “é sobre” pode ser útil o estudo das propriedades que conduzem à sua rejeição. Por exemplo o postulado

$$\frac{S \downarrow T}{S \downarrow T \cup U}$$

estabelece que se uma situação *não* é sobre outra situação, esta relação não pode surgir da adição de informação à conclusão que deve ser derivada. Este postulado apresenta um relacionamento muito próximo com o comportamento não-monotônico.

5.7 Redes Situadas

Propõe-se aqui a aplicação da lógica das situações na representação da *relevância* em uma rede de hiperdocumentos. O conceito de *rede situada* é fundamental nesta representação. Uma rede situada é uma tripla $\langle H, D, R \rangle$, onde:

- $H = \{h_1, h_2, \dots, h_n\}$ é um conjunto de hiperdocumentos;
- $D = \{d_1, d_2, \dots, d_n\}$ é um conjunto de descritores dos hiperdocumentos em H ;
- $R = \{r_1, r_2, \dots, r_m\}$ é um conjunto de regras de transição

Há uma correspondência 1-1 entre cada hiperdocumento h_i em H e os descritores d_i em D , tal que $d_i = d(h_i)$. Um descritor d_i é constituído por ínfons descrevendo individualmente diversos atributos do hiperdocumento associado h_i . Possíveis atributos do documento representados no descritor seriam: a identidade do documento, lista de palavras-chave, lista de links para os quais o documento aponta, data da última atualização, etc. As regras de transição em R , estabelecem os critérios para a evolução da rede e não se aplicam sobre os hiperdocumentos em H , mas sim sobre seus descritores em D . Estes são vistos como situações, de modo que valem para eles as relações apresentadas nas seções anteriores.

No modelo proposto empregam-se operadores de situações para estabelecer o *grau de relevância* G existente entre dois documentos a partir de seus descritores. Como os descritores são situações constituídas por um pequeno número de ínfons, previamente conhecidos e estes, por sua vez, empregam um vocabulário limitado, extraído do domínio da aplicação, torna-se possível estimar o grau de relevância G entre dois nodos A e B da rede a partir dos ínfons presentes nos descritores de ambos. Estas idéias são sintetizadas na figura 5.1.

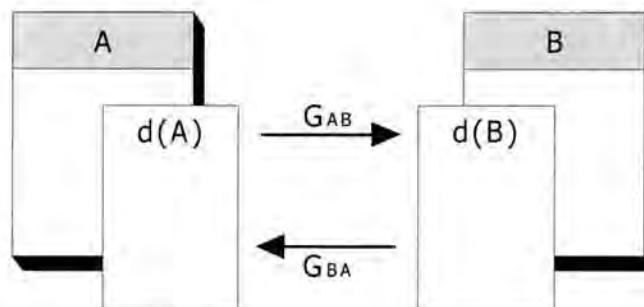


FIGURA 5.1 - Determinação do grau de relevância entre dois documentos

Em uma rede situada, os descritores d_i em D (assim como as regras r_i em R) podem ser projetados de diversas maneiras, de acordo com a aplicação ou necessidade de modelagem. A presença dos descritores no modelo permite reduzir consideravelmente o espaço de inferência e a representação destes como situações lhes permite empregar todo o aparato inferencial da lógica das situações. Uma rede situada é portanto uma rede onde cada nodo é formado por um par (documento, descriptor) e o relacionamento entre os nodos se dá através da determinação do grau de relevância G existente entre eles. Um descriptor pode assumir por exemplo o formato dado na figura 5.2, a seguir.

```

<<ident, A, 1>>
<<título, LogicWeb, 1>>
<<autor, Seng Wai Loke, 1>>
<<ult atu, 07/09/98, 1>>
<<pchave, programação em lógica, 1>>
<<pchave, prolog, 1>>
<<pchave, world wide web, 1>>
<<g, (B, 0.82), 1>>
<<g, (C, 0.17), 1>>

```

FIGURA 5.2 - Um descriptor impreciso

Nela representa-se o descriptor simplificado de um documento identificado por A . O documento intitula-se *LogicWeb*, foi escrito por *Seng Wai Loke*, a última atualização foi efetuada em 7 de setembro de 1998 e o documento possui as palavras-chave *programação em lógica*, *prolog* e *world wide web*. O valor de G está fixado em 0.82 para com o documento B , e 0.17 para com o documento C . Isto indica que B é muito relevante para o conteúdo de A , enquanto que C somente um pouco.

Uma forma simples de calcular o grau de relevância entre os nodos de uma rede situada seria aplicar o operador de composição sobre as palavras-chave nos nodos, extraindo do resultado o percentual de cobertura das palavras-chave em cada um dos descritores participantes da composição. Isto entretanto irá produzir resultados pouco precisos, a menos que as palavras-chaves estejam classificadas de acordo com sua importância na descrição do documento. Se, ao contrário, for assumida a mesma importância para todas as palavras-chaves no descriptor, há uma grande chance de que a relação de relevância estabelecida se distancie da realidade que se deseja representar.

Uma alternativa seria então estabelecer, para cada palavra-chave, um índice I representando a sua *importância* na descrição do documento ao qual está associada. Isto pode ser um processo manual ou pode ser executado por meio de ferramentas de análise automática de textos, que não somente extraem as palavras-chaves do texto como também estabelecem um escore para a sua capacidade de descrever o documento. O descriptor do documento A , apresentado na figura 5.2, passa a ter a aparência da figura 5.3.

```

<<ident, A, 1>>
<<título, LogicWeb, 1>>
<<autor, Seng Wai Loke, 1>>
<<ult atu, 07/09/98, 1>>
<<pchave, (programação em lógica, 0.27), 1>>
<<pchave, (prolog, 0.86), 1>>
<<pchave, (world wide web, 0.72), 1>>
<<g, (B, 0.82), 1>>
<<g, (C, 0.17), 1>>

```

FIGURA 5.3 - Um descritor quantizado

A precisão na determinação de G torna-se muito maior com a classificação atribuída por I ao relacionamento entre cada palavra-chave e o conteúdo do documento que ela procura descrever. Considerando agora o documento A descrito na figura 5.3 e um outro documento D cujo descritor contém os infons:

```

<<pchave, (prolog, 0.91), 1>>
<<pchave, (world wide web, 0.58), 1>>
<<pchave, (cgi, 0.78), 1>>

```

os graus de relevância entre A e D e entre D e A podem ser então calculados com maior precisão a partir da cobertura *percentual* das palavras-chave do documento pelo resultado da composição, como é mostrado na tabela 5.5

TABELA 5.5 - Cálculo do grau de relevância entre documentos

| Palavras-chave | A | D | A (norm) | D (norm) | G_{AD} | G_{DA} |
|-----------------------|------|------|----------|----------|----------|----------|
| prolog | 0.86 | 0.91 | 0.4649 | 0.4009 | 0.4649 | 0.4009 |
| world wide web | 0.72 | 0.58 | 0.3892 | 0.2555 | 0.3892 | 0.2555 |
| programação em lógica | 0.27 | - | 0.1459 | - | - | - |
| cgi | - | 0.78 | - | 0.3436 | - | - |
| Total | 1.85 | 2.27 | 1.0000 | 1.0000 | 0.8541 | 0.6564 |

Conforme a tabela 5.5, o grau de relevância de A para D é calculado pelo percentual de cobertura de A pelos índices normalizados das palavras-chaves presentes na composição $A \cap D$, o mesmo ocorrendo no cálculo do grau de relevância de D em relação a A. Note-se que G_{AD} é maior do que G_{DA} , significando que D é *mais* relevante para A do que A para D. O emprego desta técnica permite quantificar o grau de

relevância entre os nodos de uma rede situada tornando fácil sua correspondência com as redes quantizadas descritas no capítulo 4.

5.8 Sobreposição de Modelos

Enquanto que as redes quantizadas, operadas pela teoria do fecho, buscam representar os aspectos dinâmicos da navegação na web, as redes situadas, operadas pela lógica das situações, são inerentemente estáticas e tentam traduzir a noção de relevância que pode existir ou não entre os *conteúdos* dos nodos da rede. No primeiro caso o foco do modelo é a conexão. No segundo é o conteúdo semântico dos documentos, expresso em seus descritores. Uma visão da sobreposição das duas redes é dada na Figura 5.4.

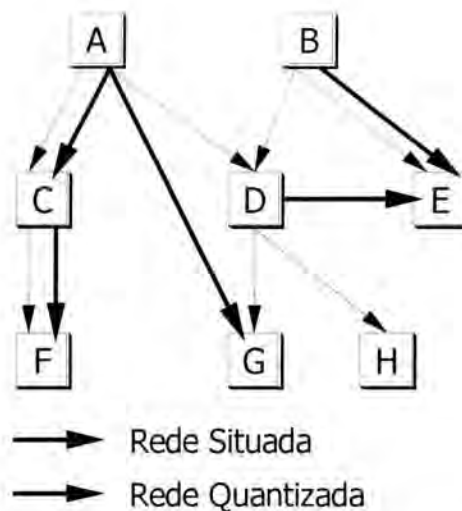


FIGURA 5.4 - Redes situadas e quantizadas sobrepostas

Na rede sobreposta, há quatro casos diferentes que podem ocorrer entre dois nodos:

1. Não há nenhuma conexão entre eles;
2. Só há conexões da rede quantizada;
3. Só há conexões da rede situada;
4. Há conexões da rede quantizada e da rede situada.

Todos esses casos podem ser convenientemente trabalhadas em um sistema de modelagem proativa. Por exemplo, se o foco estiver na rede quantizada, a existência de uma conexão situada em paralelo a um link quantizado pode reforçar o seu potencial de ativação. Similarmente a rede situada pode inferir uma conexão semântica até então inexistente entre dois nodos se entre eles houver um link quantizado com potencial de ativação suficientemente forte.

A sobreposição do modelo simbólico, representado pela lógica das situações, ao modelo conexionista da teoria do fecho em uma mesma estrutura de redes sobrepostas, está diretamente relacionada à proposta de adaptação proativa. Na situação apresentada,

cada um dos modelos preserva sua própria identidade ao mesmo tempo em que se vale da informação do outro para inferir novas possíveis conexões, reforçar ou enfraquecer conexões existentes. Estas idéias são exploradas no próximo capítulo, onde se propõe uma arquitetura para sistemas de adaptação proativa baseada na integração desses dois modelos.

Capítulo 6

Uma Arquitetura para HA Proativa

Apresenta-se neste capítulo uma proposta para a integração dos modelos introduzidos nos capítulos 4 e 5 com vistas a geração proativa de HA. Uma arquitetura genérica para a construção de sistemas de HA proativa é estabelecida e detalhada, sendo também discutidos alguns aspectos da sua implementação com o emprego de programação em lógica.

6.1 Introdução

A representação simbólica de redes de informações empregando a lógica das situações, apresentada no capítulo 5, modela uma relação de *relevância* entre dois nodos no hiperespaço considerado a partir da semântica atribuída a eles por seus projetistas. Assim as diversas páginas que compõem um site na web guardam entre si esta relação, que pode ser estendida a outros nodos por meio de operadores simbólicos como a composição.

Observa-se nesta representação uma grande facilidade para modelar relacionamentos tanto a nível conceitual quanto na esfera das situações e também entre conceitos e situações. Este entretanto é um modelo estrutural em relação ao *conteúdo* dos nodos e independe da dinâmica da navegação que é a base na inferência por fechamento. Esta segunda representação, de base conexionista, foi introduzida no capítulo 4 e trata da evolução de redes de informações a partir da aplicação dos fechos transitivo e simétrico sobre a frequência de ativação de seus links.

Ambos os modelos atuam na inferência de relacionamentos entre os mesmos nodos em uma rede, porém seus *insumos* são completamente diferentes. Na proposta que será apresentada neste capítulo busca-se explorar as características complementares dos dois modelos na construção de uma arquitetura genérica para o desenvolvimento de aplicações baseadas em hipermídia adaptativa.

6.2 Uma Arquitetura Genérica

A principal tarefa de um sistema de adaptação proativa é manter em funcionamento contínuo três processos assíncronos e até certo ponto independentes: o processo de *modelagem*, o processo de *adaptação* e o processo de *interface*. O primeiro executa a criação e manutenção de modelos de usuários. O segundo utiliza esses modelos de usuários para adaptar suas consultas e também a hipermídia recebida em resposta. Finalmente o processo de interface oferece suporte à navegação e à apresentação da hipermídia ao usuário. Na visão perseguida, os três processos se organizam segundo três eixos produzindo três planos de referência no hiperespaço:

1. interface-modelagem;
2. interface-adaptação;
3. modelagem-adaptação.

O relacionamento entre estes planos pode ser observado na Figura 6.1, onde se representa as diferentes projeções de um objeto segundo cada um deles.

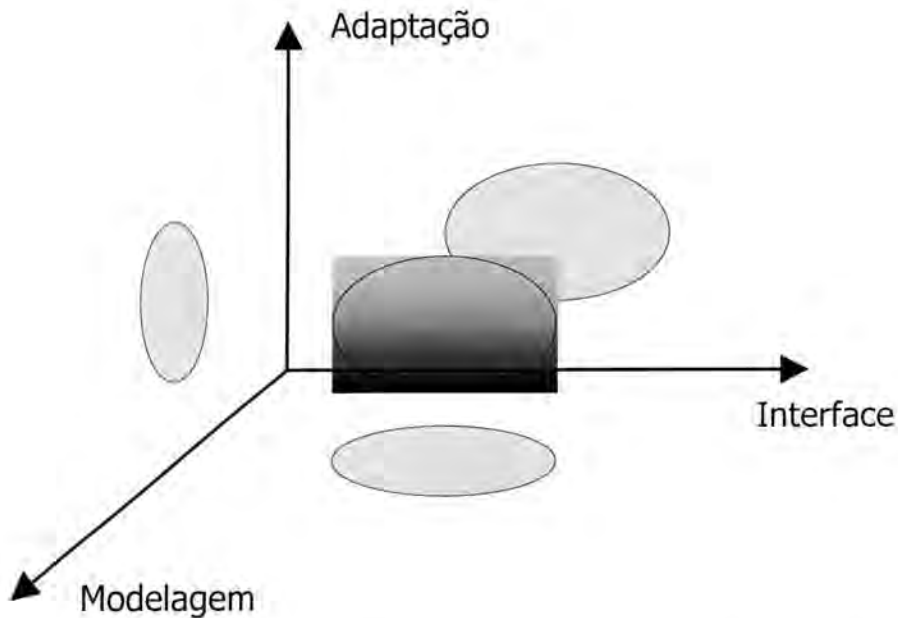


FIGURA 6.1 - Projeções de um objeto nos diferentes planos da HA.

A separação natural entre os três processos e a observação dos planos e das projeções neles produzidas conduziu à proposta de uma arquitetura orientada a agentes cujo escopo de atuação é um dos planos de projeção. A função dos agentes é estabelecer e manter a comunicação entre os dois processos negociando no sentido de oferecer a melhor tradução possível dos objetos hipermídia segundo as projeções que permitem elaborar. Uma visão mais detalhada da arquitetura do sistema é mostrada na Figura 6.2.

Na arquitetura proposta na Figura 6.2, o processo de modelagem recebe do agente de interface informação relevante sobre os diferentes usuários, produzindo seu perfil ou modelo. Tais modelos são mantidos em permanente atualização na base de modelos de usuários (bmu). Isto é obtido através de dois agentes autônomos: o agente de feedback, que extrai modelos simbólicos a partir da informação fornecida diretamente pelo usuário, e o agente de log, que extrai modelos conexionistas a partir da informação navegacional disponível sobre o comportamento do usuário em sua ação rotineira. Um terceiro agente, denominado agente de modelagem, reúne estes dois modelos e os introduz na bmu, que fornece os modelos de usuários ao processo de adaptação.

O efeito adaptativo é obtido através da execução de duas principais tarefas: (1) a adaptação das consultas segundo os modelos particulares dos usuários que as produziram e (2) a adaptação da informação recebida aos objetivos de cada usuário, conforme estabelecido em seus modelos. No conjunto espera-se que o efeito adaptativo produzido melhore em geral a recuperação de informações a partir da fonte de hipermídia considerada. O sistema é *genérico* no sentido em que o esquema de funcionamento de cada componente pode ser especificado à parte, sem interferir na sua

arquitetura. Uma visão em maior detalhe dos processos e da comunicação entre eles é apresentada nas seções seguintes.

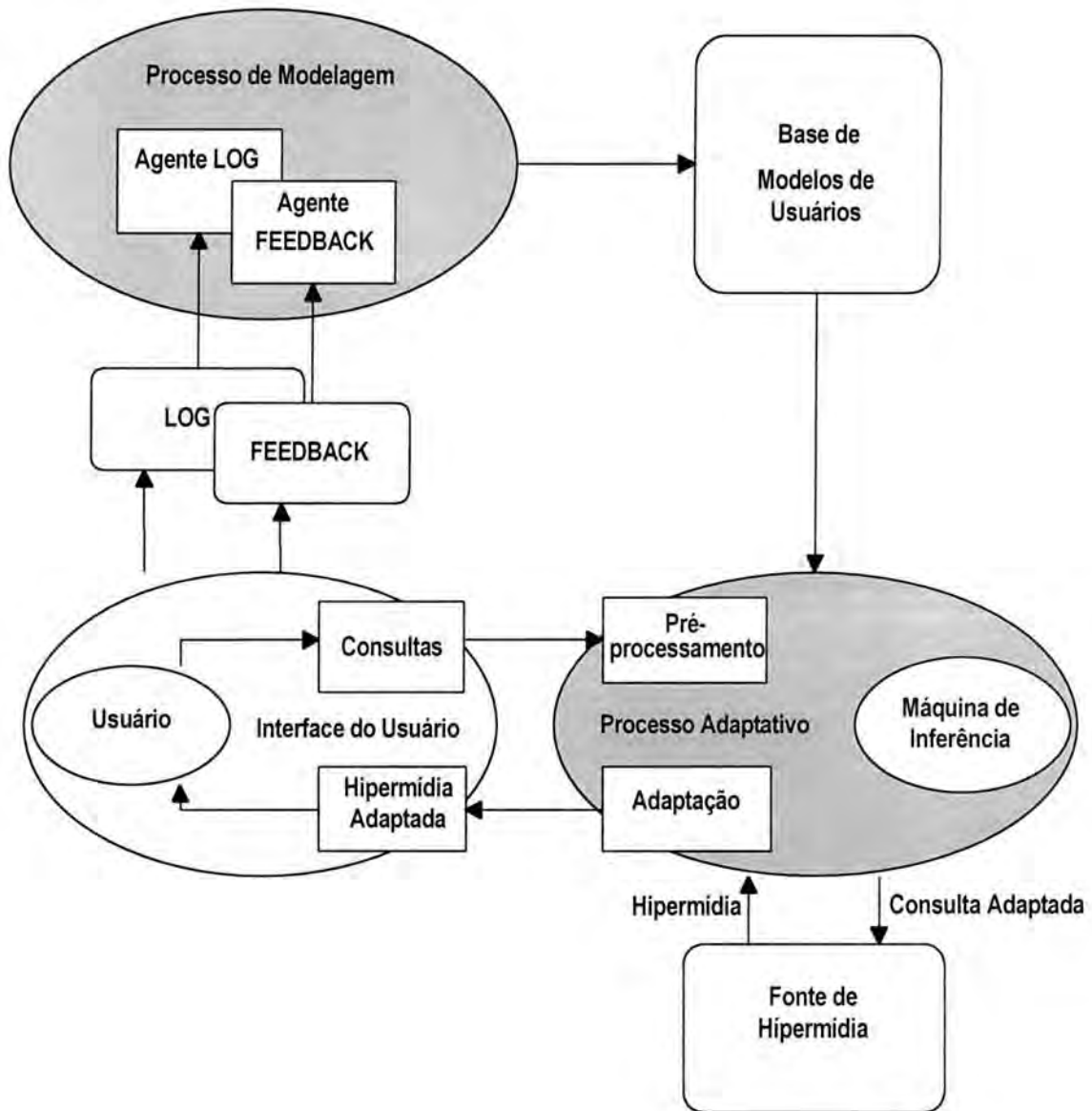


FIGURA 6.2 - Arquitetura genérica de um sistema de HA proativo

6.3 A Interface do Usuário

A interface do usuário com o sistema é representada na figura 6.3. Este componente é modelado por meio de um agente de interface que desempenha duas funções básicas:

1. Transmite as consultas e requisições originais do usuário para o agente de adaptação e recebe deste a hipermídia adaptada que será apresentada ao usuário.
2. Coleta informações de log e feedback direto do usuário, que transmite ao processo de modelagem.

Todo o contato do usuário com o sistema é realizado sob o controle do agente de interface, que idealmente é implementado no lado cliente, para execução local. A ação aqui é independente do processo de modelagem, mas está vinculada ao processo de adaptação

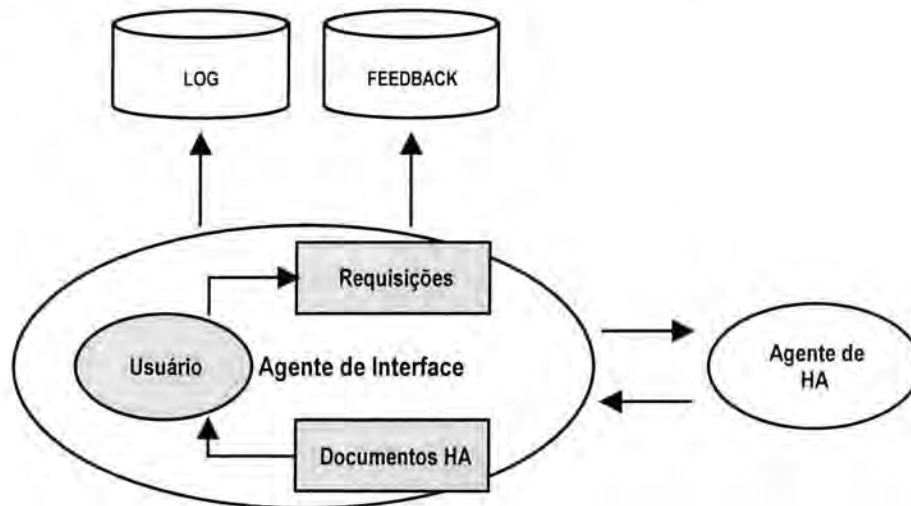


FIGURA 6.3 - A interface do usuário

Uma *consulta* ou *requisição* do usuário é a solicitação de qualquer recurso ao agente de interface. Esta solicitação é transmitida ao agente de adaptação sob a forma de uma mensagem com a especificação do recurso desejado. A solicitação pode ser uma informação, um link, um documento, software ou qualquer tipo de mídia. O agente de interface transmite e registra a solicitação em um arquivo *log*, que a intervalos regulares é remetido ao processo de modelagem.

O agente de interface também recebe a resposta do agente de adaptação (com o recurso solicitado) e providencia sua apresentação ao usuário de acordo com um conjunto de especificações previamente definidas. O sistema oferece ainda ao usuário a possibilidade de acrescentar informação ao objeto hipermídia que lhe está sendo apresentado, aprovando, rejeitando, sugerindo ou complementando seu conteúdo. Toda a informação desse tipo acrescentada pelo usuário é registrada em um arquivo *feedback*, que também é enviado periodicamente ao processo de modelagem. Esta facilidade é de grande utilidade em grupos de pesquisa, equipes de trabalho distribuídas e em sistemas educacionais online onde a colaboração entre os participantes é uma característica desejável. A geração e transmissão dos arquivos de log e feedback ao processo de

modelagem são tarefas importantes do agente de interface das quais dependem a criação e posterior atualização do modelo do usuário.

6.4 O Processo de Modelagem

Como foi visto, o processo de modelagem encontra-se a cargo de três agentes autônomos: o agente de log, o agente de feedback e o agente de modelagem. Estes três agentes produzem e mantêm atualizados os perfis ou modelos dos usuários do sistema. Tais modelos são armazenados na base de modelos de usuários (bmu), onde permanecem à disposição do agente de adaptação. A modelagem do usuário é um processo contínuo e totalmente assíncrono em relação aos processos de adaptação e interface. Seu produto final, o modelo do usuário, é construído na presente proposta a partir de duas representações básicas:

1. A rede de navegação do usuário, com os links visitados, produzida pelo agente de log, e
2. Uma coleção de atributos, palavras-chaves, objetivos, preferências e rejeições manifestadas pelo usuário, organizada pelo agente de feedback.

O processo como um todo é mostrado na Figura 6.4.

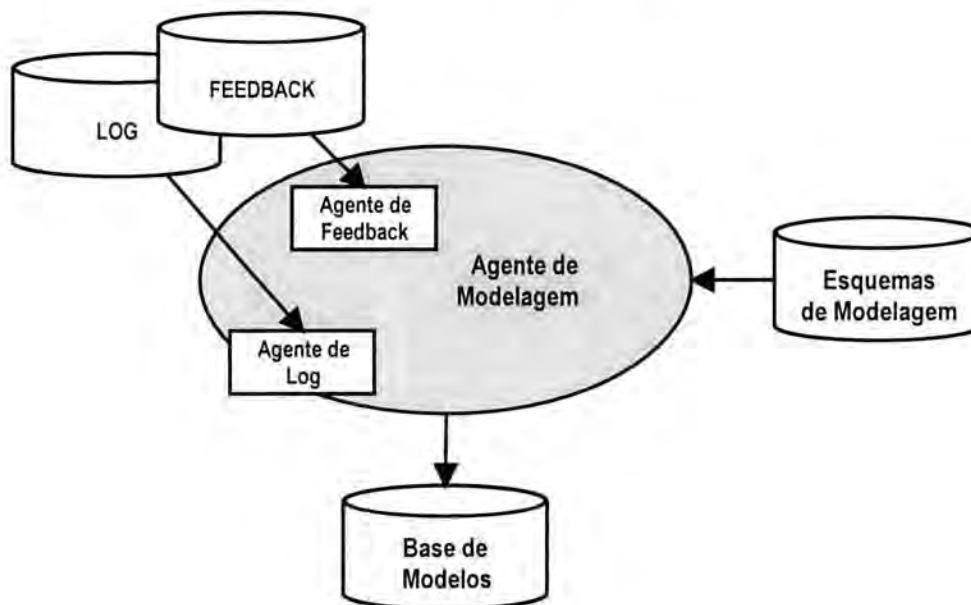


FIGURA 6.4 - O processo de modelagem

As representações produzidas pelos agentes de log e feedback são interpretadas e reunidas pelo agente de modelagem que as introduz apropriadamente na bmu. Ali as novas representações que chegam continuamente irão refletir seu conteúdo no modelo do usuário, caso este já exista, ou criar um novo modelo para usuários recém registrados. Preservando a generalidade do sistema, as atividades dos três agentes podem ser totalmente configuradas por meio de *esquemas de modelagem*, que estabelecem como as representações devem ser extraídas de seus arquivos de origem e como devem ser reunidas e introduzidas na bmu.

6.5 A Base de Modelos de Usuários

A base de modelos de usuários é a estrutura de armazenamento e recuperação onde os modelos de todos os usuários do sistema são mantidos atualizados pela ação do agente de modelagem. A bmu é um dos componentes mais importantes do sistema. Seu conteúdo são representações simbólicas e em rede que tentam traduzir diferentes aspectos do perfil de cada um dos usuários. A massa de informações que armazena, entretanto, pode ser processada de diversas maneiras, por exemplo para produzir um "modelo médio" de todos os usuários, ou modelos de subgrupos de usuários, concentrados por interesses, objetivos, etc.

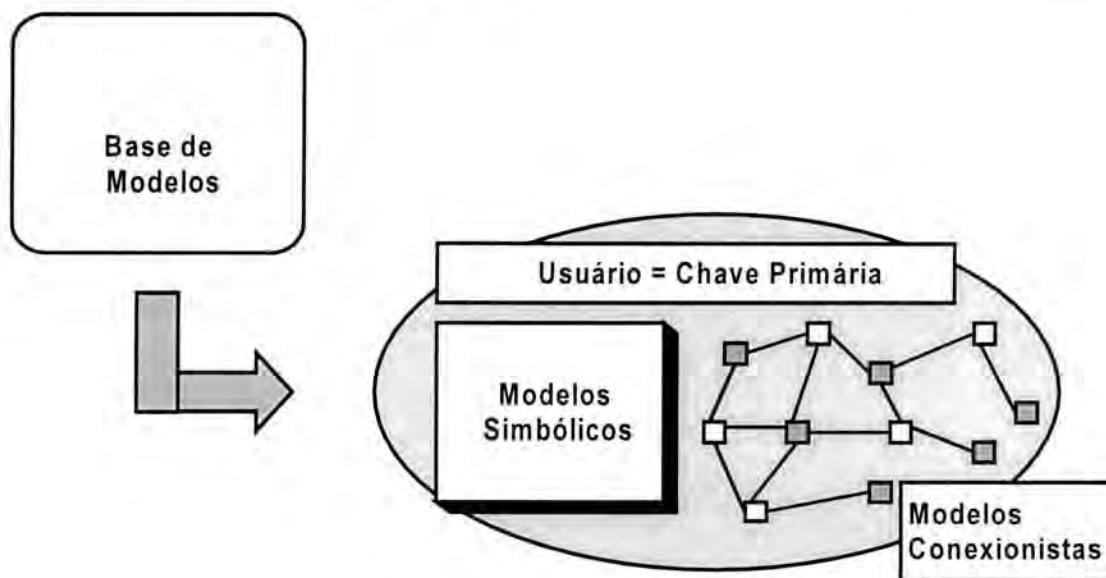


FIGURA 6.5 - O modelo do usuário

Idealmente a bmu deve permitir um esquema incremental, onde novas representações possam ser adicionadas sem prejuízo das já existentes. Novas representações são novas dimensões no espaço do modelo e devem trazer associada a semântica das relações que permitem estabelecer. O modelo de um determinado usuário se caracteriza por possuir uma chave primária (por exemplo, seu username) e dois tipos básicos de informação: (1) simbólica, de apropriação direta, sob a forma de assertivas e regras, e (2) em rede, de apropriação indireta, sob a forma de redes quantizadas, como pode ser visto na Figura 6.5. Cada uma destas representações exerce sua própria influência na construção de uma rede integrada, simultaneamente conceitual e navegacional, que em última análise é o modelo final do usuário.

Na Figura 6.6 apresenta-se a integração entre as duas representações. O relacionamento de *relevância* estabelecido entre os conteúdos dos objetos hipermedia no modelo simbólico materializa-se assim em uma rede conceitual que pode ter suas conexões quantizadas a partir do *grau de relevância* (ver capítulo 5) calculado sobre os descritores de cada nodo. Na representação da navegação, o resultado é a rede de caminhos percorridos pelo usuário no hiperespaço. Esta rede é quantizada por fechamento transitivo e simétrico sobre a frequência de ativação de seus links. As linhas pontilhadas representam relacionamentos estabelecidos entre os mesmos nodos nas duas redes que assim se integram na representação de um modelo unificado.

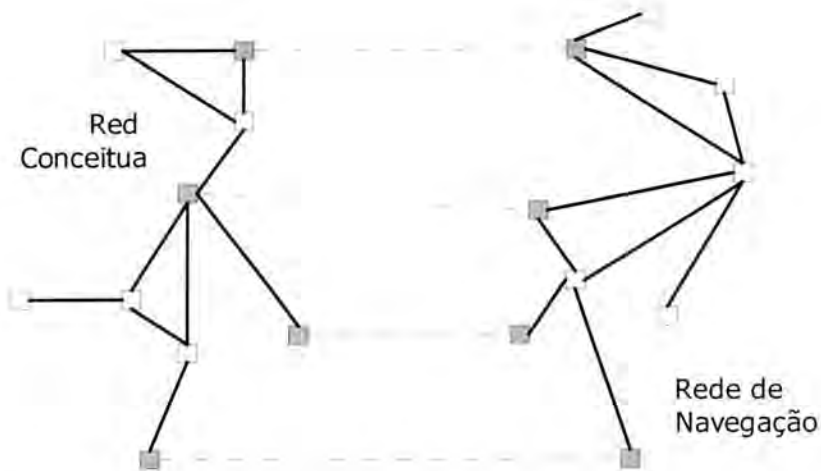


FIGURA 6.6 - Integração de representações na modelagem do usuário

6.6 O Processo de Adaptação

Este processo é responsável pela seleção e organização e configuração dos objetos hipermídia que serão apresentados ao usuário, tomando por base o seu modelo. Dada a forte coesão entre seus subprocessos, a atividade de adaptação está a cargo de um único agente que exerce múltiplas funções. A Figura 6.7 apresenta o processo de adaptação em maior detalhe.

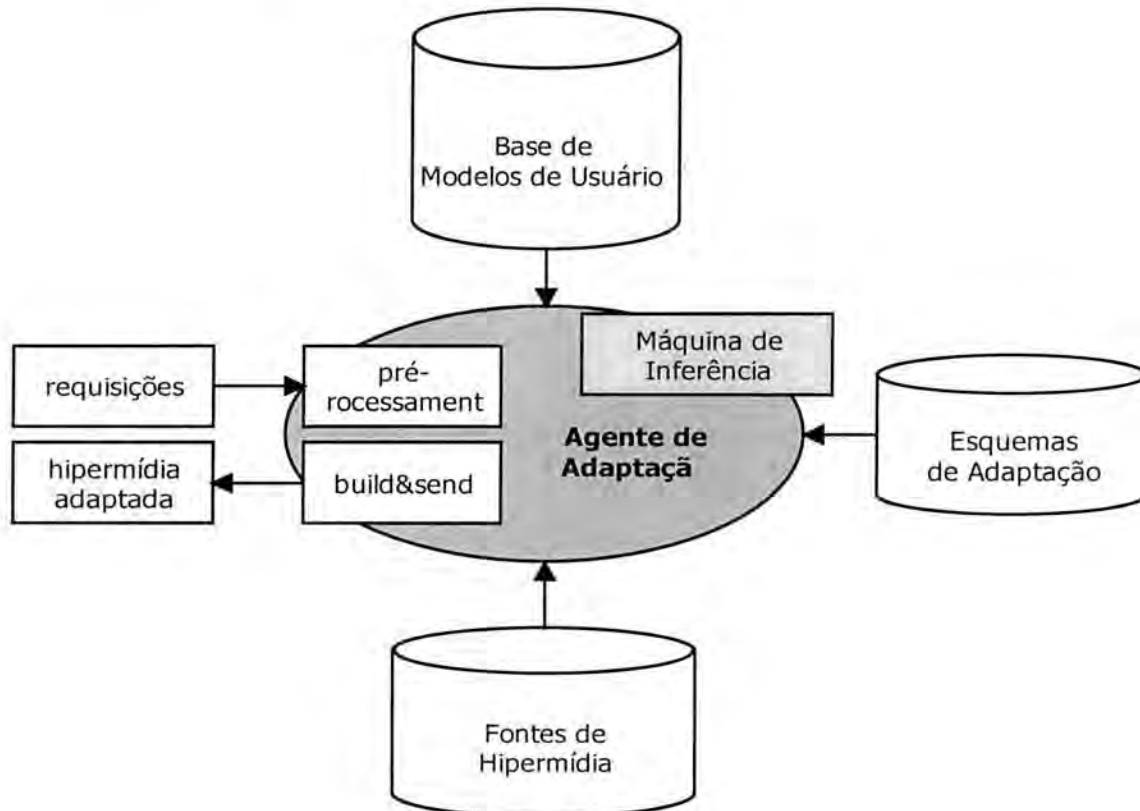


FIGURA 6.7 - O processo de adaptação

Na presente abordagem a adaptação não consiste apenas em responder de forma reativa às demandas do usuário conforme seu modelo, mas também em procurar continuamente no hiperespaço por hipermídia relevante aos objetivos manifestados no modelo do usuário. Esta ação proativa pode ser obtida por exemplo através de cópias móveis do agente de adaptação, com a função de pesquisa adaptativa sendo executada no lado cliente ou, de forma mais realista, por meio de um robô de pesquisa adaptativa operando em modo contínuo no lado servidor. Em linhas gerais o processo de adaptação consiste nas seguintes ações:

1. Recepção e pré-processamento das requisições do usuário;
2. Inferência e formulação de padrões de pesquisa no hiperespaço;
3. Seleção, classificação e envio da hipermídia adaptada ao agente de interface.

O pré-processamento das requisições do usuário enviadas pelo agente de interface consiste em identificar seu *tipo* (consulta, link, recurso, etc.) e o usuário de origem, criando as condições necessárias para a pesquisa adaptativa no hiperespaço. A máquina de inferência recebe os pares demanda-modelo e formula a partir deles um ou mais padrões de consultas para pesquisa no hiperespaço. Os resultados obtidos são selecionados e processados novamente segundo o modelo do usuário antes de serem enviados ao destinatário. Deve-se notar que a primeira etapa não necessita obrigatoriamente ocorrer, isto é, o sistema pode funcionar exclusivamente com base no modelo do usuário sem que este formule qualquer requisição. O processo é cíclico, contínuo e possui quatro entradas:

1. Esquemas de modelagem;
2. Modelo de usuários;
3. Espaço de hipermídia;
4. Requisições ou consultas

Os esquemas de modelagem estabelecem as regras e parâmetros determinantes do comportamento do agente e do processo de modelagem como um todo. Na presente proposta os esquemas de modelagem assim como os demais componentes são representados por conjuntos de sentenças lógicas especificando uma estratégia, estrutura ou configuração. Modelos de usuários representam os próprios no contexto do sistema. São mantidos atualizados na bmu de onde podem ser recarregados a intervalos regulares ou em determinados momentos para manter a cópia do sistema sempre atualizada. O espaço ou fonte de hipermídia constitui o domínio de acesso disponibilizado pelo sistema. Este componente pode apresentar diversos graus de demarcação, em geral abrangendo um subconjunto da web e outros serviços sobre a internet, como mail news e ftp.

Requisições, consultas e informações vindas do usuário constituem elementos que exigem atenção imediata e devem ser priorizados pelo sistema. Ao atender tais demandas o sistema irá gerar um processamento adicional ao que seria executado por um browser convencional como o Netscape ou o Internet Explorer. Este processamento é devido aos processos de modelagem e adaptação. A sobrecarga produzida pelo processo de modelagem, que é contínuo e executado em background, não deve chegar a ser percebida pelos usuários, entretanto a sobrecarga do processo de adaptação precisa ser avaliada cuidadosamente para não ocasionar um impacto além do razoável sobre o tempo de resposta ao usuário. Um possível paliativo seria executar parte do

processamento no lado cliente. Isto seria viável desde que o cliente possuísse uma cópia do modelo do usuário, que poderia ser transmitida pelo agente de adaptação durante o processo de identificação.

6.7 Especificação Lógica do Sistema

Adotou-se uma visão em que os diversos componentes do sistema serão representados por conjuntos de sentenças lógicas (ínfons), organizados em hierarquias onde cada nodo (página web) com suas ramificações (links) possui um descritor que corresponde a uma *situação* (contexto). Este pode se relacionar com outras situações (descritores de outras páginas) de formas múltiplas e complexas em comparação com o simples "click" disponibilizado pelo html.

Uma página web pode ser representada em Prolog por um fato:

```
página(URL, Texto).
```

onde URL é a URL da página e Texto é o seu conteúdo na forma de um string contínuo de texto, incluindo as tags html. Esta representação mantém o código html original e permite a inserção no bloco de texto de comandos Prolog entre as tags <pre> </pre>. Estes comandos podem ser interpretados e executados no lado cliente.

A alternativa adotada aqui é a representação

```
página(URL, Texto, Descritor).
```

Aqui *Texto* é o conteúdo original da página e *Descritor* um módulo explícito de assertivas (ínfons) sobre a página.. Ambas as soluções encapsulam o código html e permitem a auto-referência. (O código Prolog pode referenciar a própria estrutura onde está contido.)

A representação dos links pode ser extraída direto do texto html, pela interpretação dos tags . Por exemplo:

```
Beba <a href="http://www.cocacola.com.br" > Coca Cola </a>
```

Esta frase encontrada no texto da página irá produzir o fato:

```
link("http://www.cocacola.com.br", "Coca Cola").
```

Supondo agora um usuário fictício, o seguinte código Prolog poderia ser inserido na sua página (usando qualquer uma das duas soluções acima):

```
interesses("IA", "Redes Neurais", "Web", "Lisp").
tem_a_ver("Agentes", "IA").
tem_a_ver("Mapas de Kohonen", "Redes Neurais").
tem_a_ver("OOP", "Lisp").
páginas_úteis("Fomento",
              [
                "http://www.cnpq.br/",
                "http://www.ufpel.tche.br/~fapergs"
              ] ).
interessado_em(X):-
    interesses(Lista), membro(X, Lista).
```

interessado_em(X):-

tem_a_ver(X,Y), interessado_em(Y).

Os predicados *interesses/1* e *tem_a_ver/2* dão detalhes sobre os interesses do usuário e os tópicos com eles relacionados (Redes Conceituais, Thesauri). *páginas_úteis/2* poderia ser usado por mecanismos de pesquisa para procurar informações sobre órgãos de fomento à pesquisa. *Interessado_em/1* mostra como é possível inferir mais informações sobre os interesses do usuário. Os predicados não precisam ser qualificados com a identidade do usuário, pois estão no seu *contexto* ou *situação*.

Esta abordagem requer ver descritores de páginas web como representando situações, entre as quais há relações de maior ou menor *relevância*. Quantificar a relevância de cada relacionamento é uma tarefa que pode ser obtida com lógica, mas também a partir da observação do processo de navegação na rede. Mantido este ponto de vista, todos os demais componentes do sistema podem ser descritos como conjuntos de assertivas - módulos de programas na linguagem Prolog representando situações. O sistema de HA pode então ser representado por uma tripla:

sistema(Info, Agentes, EF), onde:

1. *Info* é o conjunto de informações ou estruturas de informações que são objeto da ação dos agentes. Na arquitetura apresentada pertencem a esta categoria os arquivos de log e feedback, a base de modelos de usuários, as fontes de hipermídia e as consultas ou requisições dos usuários.
2. *Agentes* são os componentes ativos do sistema. Sua ação é exercida sobre Info e dela depende o funcionamento do sistema como um todo. Pertencem a esta categoria os agentes de interface, de log, de feedback, de modelagem e de adaptação.
3. EF são os *esquemas de funcionamento*, conjuntos de configurações, parâmetros e regras que definem como os processos de modelagem, adaptação e interface devem ser executados pelos agentes.

Todos estes componentes possuem descritores que os representam como conjuntos de assertivas. Externamente, do ponto de vista do Prolog, são módulos, que possuem uma semântica e um tratamento conhecido.

O sistema pode ser então programado como um conjunto de componentes escritos em Prolog que serão utilizados pelos agentes para a execução dos laços de adaptação e modelagem. Como ilustração, apresenta-se no Anexo 1 a codificação de um componente que transforma a requisição original de um usuário em uma nova requisição, adaptada ao seu modelo, antes de submetê-la ao servidor hipermídia ou outro recurso.

Capítulo 7

ia@net: Um Sistema Educacional Proativo

Neste capítulo apresenta-se o projeto e detalhes de implementação de um sistema educacional com adaptação proativa, segundo o modelo e arquitetura introduzidos nos capítulos anteriores. O sistema pode ser acessado em <http://ha.ucpel.tche.br/>

7.1 Sistemas Educacionais Adaptativos na Web

Segundo [BRU 98] os sistemas educacionais adaptativos (SEA) baseados na web constituem uma classe particular de sistema que evoluiu a partir de duas outras anteriores: os sistemas tutores inteligentes (STI) e os sistemas de hipermídia adaptativa (SHA). Os problemas tradicionalmente associados ao desenvolvimento de sistemas educacionais foram estudados sob o ponto de vista dos STI [BUR 88]. Estes sistemas empregam o conhecimento disponível sobre o domínio e o estudante juntamente com técnicas de ensino flexíveis, para suportar um aprendizado individual, sendo a adaptabilidade um dos objetivos perseguidos.

Já os sistemas de hipermídia adaptativa derivam de uma pesquisa bem mais recente [BRU 96]. Estes sistemas usam diferentes modelos do usuário para adaptar o conteúdo e os links de páginas hipermídia. A educação é uma das principais áreas de aplicação de hipermídia adaptativa e diversos SHA têm sido construídos com esta finalidade nos últimos anos. Em um primeiro momento os SEA baseados na web poderiam ser vistos como simples implementações de STI ou SHA na web, entretanto o contexto da web causa um forte impacto sobre o projeto e a implementação de tais sistemas, que assim passam a caracterizar uma subclasse bem definida. Por exemplo, a maioria dos STI offline (em CD por exemplo), não emprega técnicas de hipermídia adaptativa, enquanto que quase todos os SEA baseados na web podem ser classificados tanto como STI quanto como SHA. Isto se deve ao impacto gerado pela natureza baseada em hipertexto da web [BRU 98].

Nas seções seguintes descreve-se a implementação do ia@net, um SEA baseado na web, para o estudo e pesquisa em inteligência artificial. O sistema persegue um estilo de adaptação proativa, tentando antecipar as demandas do usuário, empregando para isso os modelos de adaptação discutidos nos capítulos 4 e 5. Os componentes implementados no sistema real procuram mapear a arquitetura proposta no capítulo 6. Ao mesmo tempo o sistema foi construído para possibilitar o fácil ajuste dos processos de modelagem, adaptação e dos diversos parâmetros do sistema, permitindo grande flexibilidade na construção de diversas modalidades de cursos adaptativos.

7.2 ia@net: Uma Visão Geral

A construção do sistema ia@net (<http://ha.ucpel.tche.br/>), foi realizada a partir da coleta, seleção e classificação de recursos hipermídia disponíveis na web, cobrindo diversos tópicos na área de inteligência artificial. Estes recursos foram reunidos ao longo do tempo, somando hoje mais de 1300 itens entre links, programas, textos e arquivos, classificados em dez módulos, organizados aproximadamente segundo o livro de Stuart Russel e Peter Norvig, *Artificial Intelligence: A Modern Approach* (AIMA) [RUS 95]. O livro foi escolhido por ser quase uma unanimidade entre os professores de IA, sendo hoje adotado em mais de 500 universidades ao redor do mundo. A estrutura geral do curso é apresentada na figura 7.1.

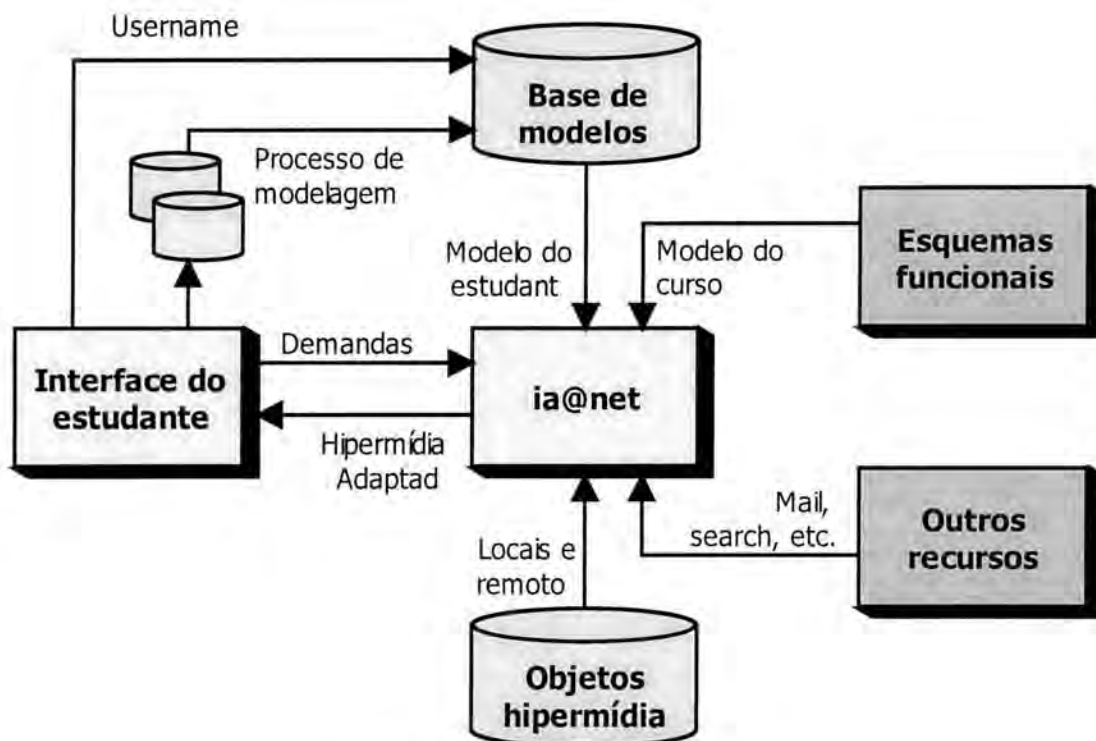


FIGURA 7.1 - Estrutura do sistema ia@net.

Conforme é mostrado na figura 7.1, o estudante é identificado pelo sistema através de um par username-password, que ocasiona a carga do seu modelo e ativa os agentes de modelagem, log e feedback (ver capítulo 6). No contexto assim constituído, as demandas do estudante em termos de material instrucional e outros recursos são canalizadas e refinadas com elementos extraídos do seu modelo, *antes* de ser transmitida ao servidor hipermídia. No retorno do recurso solicitado, este pode ser submetido à nova etapa de adaptação, a partir de elementos distintos dos utilizados na primeira etapa, antes de ser encaminhado ao estudante. O servidor e demais recursos do sistema estão instalados em uma rede cuja configuração de hardware é dada na tabela 7.1. A rede foi adquirida com recursos do CNPq (ProTeM-CC) e Fapergs, e está instalada na Escola de Informática da UCPel.

TABELA 7.1 - Configuração de hardware utilizada no projeto

| Quant. | Descrição |
|--------|---|
| 01 | Estação servidora PowerPC 604e 233Mhz, 512K cache L2, 256 Mb RAM, 27Gb Ultra Scsi Disk, 4mm tape drive, 10/100 ethernet adapter, monitor P72. |
| 04 | Pentium Celeron 300Mhz, 64Mb RAM, 8.5Gb disk, fast ethernet, CDROM 32x, monitor 15". |
| 01 | Pentium Pró 200MHz, 64Mb RAM, 6.4Gb disk, fast ethernet, CDROM 8x, monitor 15". |
| 01 | Impressora HP-5L, laser, 6ppm, 1200dpi. |

Outros recursos de hardware como scanner, captura de vídeo, impressão colorida, etc., foram disponibilizados pelo Núcleo de Pesquisa e Desenvolvimento em Informática (NAPI) da Escola de Informática da UCPel. O sistema operacional preferencialmente empregado no desenvolvimento do projeto foi Linux (Red Hat 6.0), tendo em vista a variedade e qualidade das ferramentas que esta plataforma oferece para estações de trabalho de pequeno porte. Tais recursos são distribuídos na maior parte das vezes com o código fonte e em regime *freeware*, o que os torna muito atraentes para a pesquisa acadêmica e o desenvolvimento de projetos de software em geral.

7.3 Estruturas de Modelagem Física

O sistema ia@net foi concebido tendo como um dos objetivos colher subsídios para a construção futura de uma ferramenta de autoria de sistemas educacionais adaptativos para a web. Assim, foi projetado sobre três estruturas: uma *estática*, uma *funcional* e uma *dinâmica*. A estrutura estática corresponde aos componentes físicos materializados do sistema e seus princípios básicos de organização. Exemplos de tais componentes são o esquema da base de modelos de estudantes (bme), a organização dos diretórios, o desenho básico da interface, a linguagem de comunicação entre os componentes, etc.

A estrutura funcional é provida pelos *esquemas funcionais*, que são conjuntos de regras e valores de parâmetros determinando a configuração e o conseqüente funcionamento do sistema. Estes esquemas são inicialmente determinados pelos idealizadores do sistema educacional. Posteriormente, ao longo do processo de adaptação, os estudantes passam a controlar, direta e indiretamente, diversos aspectos

do funcionamento do sistema, que se apresenta diferentemente para cada estudante, adaptando-se ao seu perfil, ritmo e preferências particulares.

Por fim, a estrutura dinâmica consiste nas seqüências de transformações a que o sistema é submetido durante seu funcionamento, incluindo aqui a atualização contínua dos modelos dos estudantes (processo de modelagem), a busca contínua por recursos hipermídia adaptados ao modelo do estudante (processo de adaptação) e a apresentação desses recursos ao estudante (processo de interface). Essas três estruturas permitem a visão do sistema segundo três diferentes dimensões, o que veio a contribuir muito para o seu entendimento e construção.

Na sobreposição das três estruturas, a estrutura estática oferece uma representação qualitativa, persistente e instantânea dos componentes do sistema e de seus atributos, recursivamente estáticos; A estrutura funcional estabelece como os objetos da estrutura estática podem ser mapeados do seu estado corrente para um possível estado futuro e também determina os espaços de variação para os valores dos parâmetros e demais restrições impostas ao sistema e a seus componentes. A estrutura dinâmica reflete o que *realmente acontece*: das muitas possibilidades de estados, qual irá ocupar o momento seguinte na seqüência de transições?

As três estruturas situam-se em diferentes graus de abstração, como pode ser visto na figura 7.2, onde é possível identificar a restrição progressivamente exercida primeiro pela estrutura funcional sobre as possibilidades de transição entre estados da estrutura estática e da estrutura dinâmica, tomada como registro da realidade em relação às possibilidades admitidas pela estrutura funcional. No presente nível de abstração os detalhes de construção do sistema ainda não são aparentes, mas é possível conceber desde já que cada componente do sistema irá se relacionar de forma distinta com cada uma das três estruturas apresentadas.



FIGURA 7.2 - As estruturas estática, funcional e dinâmica em diferentes graus de abstração

7.4 O Modelo Pedagógico

O objetivo do sistema ia@net é disponibilizar na web, através de adaptação proativa, um grande número de recursos associados à IA, que possam ser empregados na educação formal (disciplinas e cursos) ou informal (estudos, pesquisas e referência). Este objetivo, genérico e abrangente, é possível uma vez que a arquitetura do sistema foi concebida de modo a isolar o esquema de funcionamento dos demais componentes, permitindo assim múltiplas configurações, adaptadas à situação que se venha a desejar.

7.4.1 O Modelo Educacional

Na atual configuração, o ia@net é um curso de auto-estudo, onde o estudante escolhe o programa que deseja cursar (dentre vários possíveis), seleciona os conteúdos que lhe parecerem mais interessantes em cada módulo e sobre eles realiza *tarefas*, produzindo através delas novos conhecimentos *relevantes* que serão disponibilizados aos demais estudantes do sistema. Em cada módulo o estudante é estimulado a avaliar os recursos que lhe estão sendo apresentados e se submete a uma auto-avaliação criteriosa, antes de ser considerado aprovado. Dos dez módulos oferecidos, são obrigatórios apenas os módulos 1, 2 e 10. Dos sete módulos restantes o estudante deve escolher três, totalizando seis módulos que correspondem a um programa de curso.

Todos os alunos inscritos registram sua conta de e-mail no sistema e são inscritos automaticamente em duas listas de discussão. A primeira possui um caráter geral e nela os estudantes podem trocar idéias e opiniões sobre IA, sobre o sistema ia@net e outros assuntos relacionados. Os estudantes permanecem obrigatoriamente inscritos nesta lista enquanto estiverem vinculados ao curso. Através dela, além de assuntos relacionados com IA, os estudantes irão receber avisos, sugestões, informações, etc. A segunda lista é específica do módulo que o estudante estiver cursando. Nesta lista são discutidos exclusivamente os aspectos da matéria ali estudada. Ambas as listas são moderadas e constituem um importante meio de comunicação entre os estudantes e os instrutores. Novos estudantes têm acesso às mensagens anteriores, que são armazenadas e classificadas em ordem cronológica, por assunto e linha de desenvolvimento.

A parte proativa neste projeto fica por conta de um robô de pesquisa, que sistematicamente busca na web por material relevante para o modelo do usuário e o assunto que este está estudando. O robô recebe o modelo do usuário e o do módulo que este está estudando sob a forma de um conjunto de palavras-chave e tenta localizar no hiperespaço documentos e outros recursos cujos descritores permitam estabelecer um grau de relevância elevado para com estes modelos (ver capítulo 5). Os links assim obtidos são classificados e apresentados ao estudante para avaliação. Descobrir estes links é um processo prolongado e o estudante não deve esperar por resultados já na primeira sessão, entretanto, sob o controle do servidor, o robô continuará pesquisando mesmo enquanto o estudante estiver desconectado do sistema, armazenando os resultados para apresentação posterior. O processo é contínuo e os novos links encontrados são adicionados a uma lista de links pessoais que o estudante pode editar incluindo, excluindo ou anotando links.

7.4.2 A Fonte de Hipermissão

A base de conteúdos (domínio) do curso é uma coleção aberta de documentos hipermissão em constante crescimento e organizaço. Esta coleço, hoje com cerca de 1300 itens entre links, programas, artigos, documentos e outros recursos, foi reunida ao longo do tempo, catalogada e classificada em dez captulos ou *mdulos*, cobrindo as dez principais sub-reas da IA, conforme se apresenta na tabela 7.2.

TABELA 7.2 - Classificaço dos conteúdos do ia@net

| MDULOS | CONTEDOS |
|---|--|
| Mdulo 1: Introduço. Conceitos Bsicos | Este mdulo procura situar a Inteligncia Artificial (IA) como cincia em funço de seus conceitos e objetivos essenciais. Apresenta as diferentes abordagens adotadas no estudo da IA e o seu relacionamento com outras cincias, como filosofia, psicologia, lingstica, etc. Relata aspectos do desenvolvimento da IA ao longo do tempo e estabelece o estado da arte vigente no final do sculo XX. |
| Mdulo 2: Agentes Inteligentes e Sistemas Multiagentes | Este mdulo estuda o conceito de agente inteligente e como este deve atuar em seu ambiente. Qual o mapeamento ideal entre percepço e ao para se atingir a autonomia de um agente. Qual a estrutura propriedades e a classificaço dos agentes. Como diversos agentes compartilham um mesmo ambiente. Como é possvel desenvolver agentes de software para atuar em ambientes virtuais. |
| Modulo 3: Estratgias de Pesquisa em Espaços de Estados | Aqui se apresenta a soluço de problemas atravs de pesquisa em espaços de estados. Aspectos da formulaço de problemas e sua representaço em espaços de estados so considerados, assim como diferentes estratgias de pesquisa (em amplitude, em profundidade, de custo uniforme, etc.), suas variantes e aplicaçes. Mtodos de pesquisa com informaço so analisados, assim como o emprego de heursticas e tcnicas interativas na soluço de problemas. Conclui-se o mdulo com o estudo de aplicaçes na rea de jogos e simulaço. |
| Mdulo 4: Representaço e Raciocnio | Estuda-se como construir agentes baseados em conhecimento. Modelos de representaço e inferncia. A lgica de primeira ordem na representaço de conhecimento. A construço de uma base de conhecimento. Inferncia em lgica de primeira ordem e sistema de raciocnio lgico, incluindo raciocnio para frente e para trs. Sistemas de programaço em lgica, sistemas de produçes redes semnticas e sistemas de frames. |
| Mdulo 5: Modelos Conexionistas | Aborda-se o emprego de perceptrons, redes neurais e outras tcnicas alternativas de base conexionista na soluço de problemas de otimizaço, adaptaço, etc. Estruturas de redes neurais. Sistemas ART e Mapas de Kohonen. Mtodos e aplicaçes. Sistemas de programaço evolutiva. Algoritmos genticos. Autmatos Celulares. Vida Artificial. Complexidade Caos e Auto-organizaço. |

| MÓDULOS | CONTEÚDOS |
|---|---|
| Módulo 6: Conhecimento Incompleto e Incerteza | Neste módulo são abordados conceitos, métodos e técnicas de planejamento autônomo, incluindo decomposição hierárquica, restrições, medidas de planejamento, etc. Estuda-se também sistemas de planejamento e ação, planejamento condicional, a construção de um agente de planejamento, integração de planos, etc. |
| Módulo 7: Elaboração de Planos | Como um agente pode atuar sob incerteza. Probabilidade. O teorema de Bayes. Sistemas de raciocínio probabilístico. Representação de conhecimento em domínios incertos. Redes de crença. Inferência em redes de crença. Tomada de decisões simples. A teoria da utilidade. Redes de decisão. O valor da informação. Tomada de decisões complexas. O projeto de agentes de decisão. Redes de decisão dinâmicas. |
| Módulo 8: Aprendizado | Um modelo geral de agente que aprende. Conceitos relacionados com aprendizado. Aprendizado Indutivo. Aprendizado em árvores de decisão. O emprego da teoria da informação no aprendizado. Aprendizado a partir de descrições lógicas. Uma teoria de aprendizado computacional. Aprendizado em redes neurais. Aprendizado em redes de crença. Métodos bayesianos para aprendizado em redes de crença. Aprendizado por reforço. Aprendizado com algoritmos genéticos e programação evolutiva. O papel do conhecimento no aprendizado. |
| Módulo 9: Comunicação, Percepção e Ação | Estudo dos agentes que se comunicam. Comunicação como ação. Classificação dos agentes de comunicação. Gramáticas formais (uma gramática formal para um subconjunto do inglês). Gramáticas de cláusulas definidas. Gramáticas aumentadas. Interpretação semântica e pragmática. Ambigüidade. Processamento da Linguagem Natural (PLN). Aplicações práticas do PLN. Percepção visual e processamento da imagem. Robótica: aplicações, composição, arquitetura, espaço de ação, movimentação e planejamento. |
| Módulo 10: Novas Aplicações | Novas aplicações: IA e modelos de desenvolvimento social. Artes e ciências. Educação. Aprendizado, criatividade e inovação. IA e a economia globalizada. Aplicações em medicina. Comunicação. Modelos biológicos e ambientais. Descoberta de conhecimento. |

O esquema de acesso à base de conteúdos foi projetado para aproveitar a própria estrutura de diretórios do sistema operacional. Segundo este esquema, cada usuário é mapeado a um único diretório, onde reside seu *modelo*. Este é constituído por múltiplas estruturas de representação que descrevem seus objetivos, conhecimento, experiência e preferências, com influência direta no resultado da adaptação.

Material hipermídia sobre cada módulo é disponibilizado ao estudante local e remotamente. A maior parte da área de armazenamento disponível para o sistema foi configurada como uma cache de grandes dimensões para replicar localmente todas as páginas acessadas remotamente. Esta medida confere grande agilidade ao sistema e assegura a disponibilidade da página solicitada.

7.4.3 A Estrutura de um Módulo

Um módulo é uma estrutura composta por três objetos:

1. Um *descriptor* contendo o título, a ementa, palavras-chave e outras características relevantes do módulo;
2. Um *roteiro*, contendo instruções e tarefas que os estudantes devem cumprir para completar o módulo;
3. Uma página de *recursos*, contendo links para diversos recursos na web relacionados com o módulo.

Descritores são componentes comuns a todos os nodos da rede. Os ínfons que os constituem são implementados na prática como cláusulas da linguagem Prolog com a respectiva semântica declarativa associada. A lógica das situações é implementada sobre esta camada de abstração, que é suficientemente elevada para permitir uma correspondência muito próxima com o significado pretendido. Um exemplo de descriptor para um módulo é apresentado na figura 7.3.

```
<<ident, Módulo 1, 1>>
<<título, Introdução e Conceitos Básicos, 1>>
<<autor, Luiz A M Palazzo, 1>>
<<ult atu, 06/11/1999, 1>>
<<pchave, (Inteligência Artificial, 0.88), 1>>
<<pchave, (IA+[introdução, fundamentos, conceitos], 0.92), 1>>
<<pchave, (IA+[filosofia, psicologia, lingüística], 0.45), 1>>
<<pchave, (IA+[informática, computação, web], 0.74), 1>>
<<g, (Módulo 2, 0.89), 1>>
<<g, (Módulo 10, 0.35), 1>>
```

FIGURA 7.3 - Um descriptor para um módulo do curso

O descriptor mostrado na figura 7.3 apresenta uma variação sintática para indicar o *contexto*. Assim a expressão *IA+[filosofia, psicologia, lingüística]* indica que filosofia, psicologia e lingüística são palavras-chave somente na presença (ou no contexto) de "IA". Extensões deste tipo foram introduzidas em alguns pontos da representação, porém em caráter meramente sintático, sem prejuízo do significado desejado..

O *roteiro* é o instrumento de que o estudante se vale para determinar e vencer seus objetivos de estudo no módulo. Fisicamente o roteiro é um documento que estabelece as metas de aprendizado do módulo e as opções que o estudante tem para cumpri-lo. Um exemplo de roteiro é dado na figura 7.4, a seguir:

Módulo Um

Introdução e Conceitos Básicos

[Objetivos do Módulo](#)
[Leituras Requeridas](#)
[Leituras Complementares](#)
[Exercícios e Tarefas](#)
[Feedback](#)
[Relatório](#)
[Palavras-chave](#)

Objetivos do Módulo

Este módulo tem por objetivo situar a Inteligência Artificial (IA) como ciência em função de seus conceitos e objetivos essenciais. Apresenta as diferentes abordagens adotadas no estudo da IA e o seu relacionamento com outras ciências, como filosofia, psicologia, lingüística, etc. Relata aspectos do desenvolvimento da IA ao longo do tempo e estabelece o estado da arte vigente no final do século XX.

Leituras Requeridas

Para completar o presente módulo é requerida a leitura do capítulo 1 do livro de Russell e Norvig: [Artificial Intelligence: a Modern Approach \(AIMA\)](#). O livro pode ser adquirido online no site da [Amazon](#). Enquanto isso, nas bibliotecas de cursos de computação há em geral um ou mais exemplares. Você também pode consultar a [tabela de conteúdos](#) do AIMA e tentar preencher os tópicos com material alternativo, encontrado na web ou a partir da página de [recursos de IA online](#). Isto é mais difícil, leva mais tempo e demanda perseverança, mas pode produzir resultados interessantes.

Leituras Complementares

Os documentos abaixo são complementares e servem para preencher possíveis lacunas em seu conhecimento geral de IA. Podem ser lidos/estudados antes, durante ou depois da leitura do capítulo 1 do AIMA e devem contribuir para a visão geral que se espera desenvolver da IA neste módulo.

[What is AI](#)
[AI FAQ](#)
[Intro to AI](#)

Uma atividade estimulada aqui é a navegação na página de [recursos de IA online](#), onde há uma grande variedade de material interessante.

Exercícios e Tarefas

Atenção: Somente execute as tarefas abaixo após a realização das leituras requeridas. As respostas solicitadas por escrito devem ser enviadas por e-mail em attach [aqui](#) ou a qualquer hora para mod1@ha.ucpel.tche.br

Tarefa 1: Pesquise na Internet e faça o download de um programa de IA à SUA escolha (p.ex: um sistema multiagente, um provador de teoremas ou um simulador de algoritmos genéticos). Escreva um relatório de uma ou duas páginas com as características técnicas e operacionais do programa, suas impressões sobre o seu desempenho e possíveis aplicações do mesmo.

Tarefa 2: Escrever um ensaio de uma página com a SUA opinião sobre o papel da IA no século XXI.

Tarefa 3: Navegue pelo site do curso, especialmente pelos links das páginas de recursos de IA online. Ao longo da sua navegação selecione, anote e envie em [feedback](#) o título e as respectivas URLs dos cinco sites que você considera os melhores. Adicione algum comentário sobre cada site.

Tarefa 4: Participe da lista de discussão do módulo. Sugira, debata, contribua, dê a sua opinião. Suas intervenções bem colocadas na lista influem positivamente na sua avaliação.

Tarefa 5: Após cumprir as tarefas anteriores, preencher o [relatório de encerramento](#). (Indispensável para avançar para o próximo módulo).

Feedback

Para remeter feedback clique [aqui](#).

Relatório de Encerramento

Cumpriu todas as tarefas do módulo? Então clique [aqui](#) para planejar seu [relatório de encerramento](#). A apresentação deste relatório é requisito indispensável para concluir o módulo.

Palavras-chave

Inteligência artificial, [introdução, fundamentos, conceitos], [filosofia, psicologia, lingüística], [informática, computação, web]

© 1999

← ↑ → | [INDICE](#) | [HOME](#)

FIGURA 7.4 - O roteiro de um módulo

Como pode ser observado no exemplo da figura 7.4, o roteiro é simplesmente um guia de atividades para o estudante e pode ser planejado de diferentes maneiras, dependendo do modelo educacional que se pretende adotar. Neste exemplo o estudante necessita realizar um conjunto de leituras *requeridas*, que podem ser complementadas com outras leituras de interesse para absorver um conjunto de noções, idéias e conceitos básicos sobre IA em geral. Além disso solicita-se que o estudante execute uma série de tarefas com o objetivo de adquirir experiência e consolidar seus conhecimentos sobre os conceitos recém adquiridos. O estudante tem ainda sua participação estimulada em debates sobre os temas do módulo com outros estudantes através da lista de discussão do módulo. Acredita-se que tal conjunto de recursos possa vir a ter maior impacto sobre o aprendizado do que é obtido normalmente em sistemas educacionais presenciais.

Além do descritor e do roteiro, cada módulo possui uma *página de recursos* reunindo links para diversos locais na web cujo tema se relaciona com os seus conteúdos. No Anexo 2 apresenta-se como exemplo a página de recursos online do módulo 1 (Introdução e Conceitos Básicos) do sistema ia@net.

7.5 Apresentação do Sistema e o Processo de Login

Ao ser acessado, através da URL <http://ha.ucpel.tche.br/>, o sistema se apresenta ao visitante com a aparência da página na figura 7.5. Ali o visitante recebe um panorama geral dos recursos disponíveis e é informado sobre o que deve esperar encontrar em cada link da página por meio de um *out* acionado pelo deslocamento do mouse. Para seguir adiante, entretanto, é necessário executar um processo de *login*.

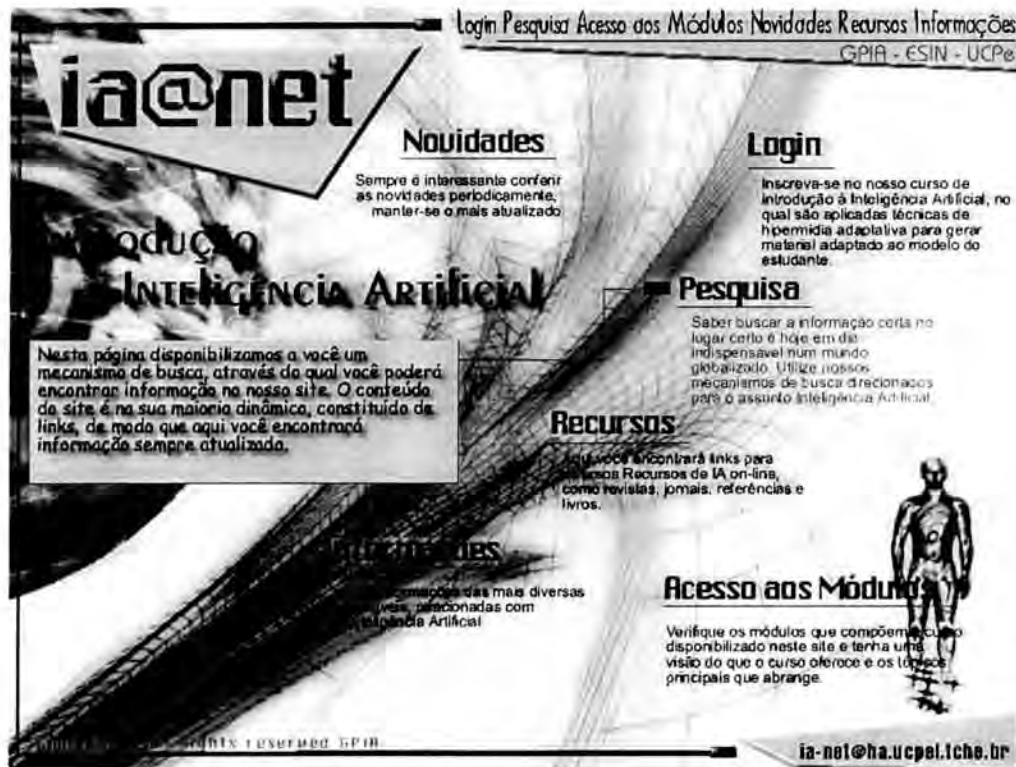


FIGURA 7.5 - A página de apresentação do sistema

O login com username/password é necessário para que o sistema consiga identificar o estudante e carregar o seu modelo, que passa então a integrar o contexto de adaptação. Caso o visitante não esteja inscrito no sistema, tem a possibilidade de realizar sua matrícula e iniciar imediatamente o estudo adaptativo de IA. Ou então a de logar-se com o username *visitante* e password *visitante*. Um visitante tem livre acesso a todos os recursos do sistema, entretanto não goza da sua capacidade adaptativa, uma vez que não há modelo para balizar a adaptação. Caso a opção seja pela matrícula, o estudante recebe um formulário online que deve ser integralmente preenchido a partir do qual é extraído o modelo inicial do estudante.

7.6 Modelagem do Estudante

O modelo do estudante (ME) é um componente fundamental em qualquer sistema educacional adaptativo e é planejado para permitir a identificação e representação precisa de diversos atributos do estudante que são relevantes para o processo de adaptação. Ao se inscrever no sistema o estudante preenche um formulário de matrícula online, do qual é extraído seu modelo inicial. Internamente este modelo é representado por um conjunto de ínfons que estabelecem uma rede conceitual, quantizada que modela simbolicamente o estudante. O componente conexionista do

modelo do estudante é obtido a partir da observação dos padrões de navegação que este realiza. Ao contrário do que se poderia pensar a princípio, o sistema não tenta atribuir um significado a tais padrões, mas os identifica e utiliza simplesmente como um padrão, para possíveis inferências futuras.

O ME é portanto construído a partir de dois componentes, um simbólico e um conexionista. *Ambos* são, entretanto, representados aqui como conjuntos de ínfons (situações), que incorporam *duas* representações em rede: a rede simbólica dos conceitos e seus graus de relevância e a rede associativa dos links e seus potenciais de ativação. Esta amalgamação de modelos se dá na apenas na camada sintática e é muito útil para permitir o acoplamento entre o componente simbólico e o conexionista.

Um exemplo de ME inicial, construído exclusivamente a partir das informações extraídas do formulário de matrícula pode ser visto na figura 7.6. Ali as informações iniciais sobre o estudante estão representadas sob a forma de ínfons e permitem recuperar o nome do estudante, sua data de nascimento, sexo, estado civil, profissão e conjuntos de palavras-chave relacionadas com os assuntos de interesse assinalados pelo estudante em seu formulário de matrícula. Sintaticamente tais palavras são representadas entre chaves (p.ex: {Notícias}), o que significa que estão denotando não apenas a si próprias, mas todo um conjunto de palavras-chave que se relacionam com elas na rede conceitual do sistema.

```
<<username, abc, 1>>
<<nome, Abel Bartolomeu Camargo, 1>>
<<email, abc@provedor.com.br, 1>>
<<nascimento, 06/11/1979, 1>>
<<sexo, masculino, 1>>
<<ecivil, solteiro, 1>>
<<profissão, (informática), 1>>
<<pchave, ({aplicações e sistemas especialistas}, 0.81), 1>>
<<pchave, ({processamento da linguagem natural}, 0.70), 1>>
<<pchave, ({notícias}, 0.54), 1>>
<<pchave, ({economia}, 0.62), 1>>
```

FIGURA 7.6 - O Modelo Inicial do Estudante

Na medida em que o estudante interage com o sistema, seu modelo vai se enriquecendo e refinando com novos ínfons, descrevendo, por exemplo, o progresso do seu aprendizado, seus hábitos de navegação, suas preferências, etc. Esta evolução do ME ocorre pela construção e desenvolvimento nele de duas redes:

1. *Rede Situada*: É a rede conceitual estabelecida com base na teoria da situação e no grau de relevância G (ver Capítulo 5) existente entre os conceitos que representam o estudante;

2. *Rede Fechada*: É a rede produzida pela aplicação das regras de transitividade e simetria da teoria do fecho sobre o potencial de ativação P (ver capítulo 4) estabelecido para os links da rede.

As representações dessas duas redes no modelo do estudante obedecem à sintaxe usual baseada em ínfons. Além disso há um mapeamento direto entre elas: os nodos da rede situada são conceitos que descrevem os nodos da rede quantizada. Estes nodos são documentos web na rede quantizada, mas aqui seu conteúdo é completamente abstraído, centrando-se a modelagem na dinâmica das conexões. Já na rede situada, as conexões são estabelecidas tendo em vista as descrições do *conteúdo* dos nodos e são geradas pela determinação do grau de relevância entre eles. Na figura 7.7 apresenta-se uma possível evolução do ME apresentado na figura 7.6, após o usuário haver interagido com o sistema durante algum tempo. Note-se o surgimento dos ínfons rs (rede situada) e rf (rede fechada), com os respectivos valores de referência correspondentes ao grau de relevância G e o potencial de ativação P.

```

<<username, abc, 1>>
<<nome, Abel Bartolomeu Camargo, 1>>
<<email, abc@provedor.com.br, 1>>
<<nascimento, 06/11/1979, 1>>
<<sexo, masculino, 1>>
<<ecivil, solteiro, 1>>
<<profissão, (informática), 1>>
<<pchave, ({aplicações e sistemas especialistas}, 0.81), 1>>
<<pchave, ({processamento da linguagem natural}, 0.70), 1>>
<<pchave, ({notícias}, 0.54), 1>>
<<pchave, ({economia}, 0.62), 1>>

<<rs, ([notícias, economia], 0.93), 1>>
<<rs, ([sistemas especialistas, economia], 0.85), 1>>
<<rs, ([processamento da linguagem natural, economia], 0.67), 1>>
<<rs, ([aplicações, informática], 0.75), 1>>

<<rf, ([http://ia.ucpel.tche.br/, http://ha.ucpel.tche.br/], 0.76), 1>>
<<rf, ([http://ia.ucpel.tch.br/, http://ai.cs.mit.edu/], 0.57), 1>>
<<rf, ([http://ia.ucpel.tche.br/, http://ah.csd.tue.nl/], 0.49), 1>>

```

FIGURA 7.7 - O Modelo do estudante após alguma interação com o sistema

Estendido com os *infos rs* e *rf* o ME ganha em abrangência de definição e também em precisão, permitindo uma maior aproximação dos objetivos de aprendizado do estudante. Essas duas redes evoluem dinamicamente ao longo do tempo e a influência que uma exerce sobre a outra pode ser definida de muitas maneiras. É fácil por exemplo imaginar um ajuste gerado simbolicamente pela composição do ME com os descritores dos documentos que são *nodos* na *rf*. Tais descritores podem ser manual ou automaticamente gerados por meio de ferramentas apropriadas de extração de conhecimento.

O acoplamento entre as duas redes pode ocorrer de diversas maneiras. Por exemplo, se a composição da *rs* no ME com o descritor de um documento da *rf* apresentar um grau de relevância elevado, então provavelmente a vizinhança quantizada do documento em *rf* será de alguma forma relevante para o estudante. Em outro sentido, a existência de um elevado potencial de ativação entre duas páginas poderia fortalecer a relação de relevância entre os conceitos presentes em seus descritores. Assim há um relacionamento sinérgico entre as duas redes que pode ser aproveitado na evolução proativa do ME.

De acordo com a arquitetura proposta no Capítulo 6, os ME são armazenados na *base de modelos de estudantes* (*bme*). Na atual versão do sistema a *bme* foi projetada visando aproveitar a estrutura hierárquica dos diretórios do próprio sistema operacional. Assim cada estudante tem o seu modelo residindo em um diretório designado pelo seu *username*, onde também são armazenados os seus trabalhos, seus arquivos de parâmetros, de log e de status, resultados de pesquisas e outros documentos de seu interesse.

Em se tratando de um protótipo, a necessidade de armazenamento local, junto ao servidor, é muito grande, tendo em vista o quanto isto facilita a realização de experimentos, simulações, o registro da atividade do sistema e sua administração. Entende-se entretanto que em versões futuras o lado cliente da aplicação possa vir a ser melhor explorado, assumindo por exemplo o controle do processo de modelagem do estudante.

7.7 Implementação do Processo de Adaptação

Um elemento importante no projeto do sistema *ia@net* é que este possui uma natureza inerentemente adaptativa, caracterizada pela flexibilidade de configuração que oferece, permitindo a definição de uma grande variedade de esquemas de modelagem e adaptação. Diversos parâmetros fundamentais do sistema, como o limiar de visibilidade das conexões e a taxa de degradação que deve ser aplicada a cada ciclo sobre o potencial de ativação de cada link, podem ser facilmente configurados no *ia@net*, *adaptando-o* a diferentes objetivos educacionais e de pesquisa.

O processo básico de adaptação, que traduz a rede de recursos hiper-mídia do sistema para um determinado ME, foi implementado em Prolog segundo as técnicas de adaptação proativa apresentadas no Capítulo 6. O nível mais elevado desta implementação é a cláusula:

```
adapta (H, ME, HA) ←
    adaptaçãoSituada (H, ME, HS),
    adaptaçãoFechada (H, ME, HF),
```


combina (HS, HF, HA).

A cláusula mostra que há duas linhas de adaptação cujos resultados são combinados para produzir o efeito adaptativo desejado. Como subproduto deste processo, são produzidas informações que alimentam o processo de modelagem, completando o ciclo adaptativo do sistema. Uma visão geral do processo é dada na figura 7.8

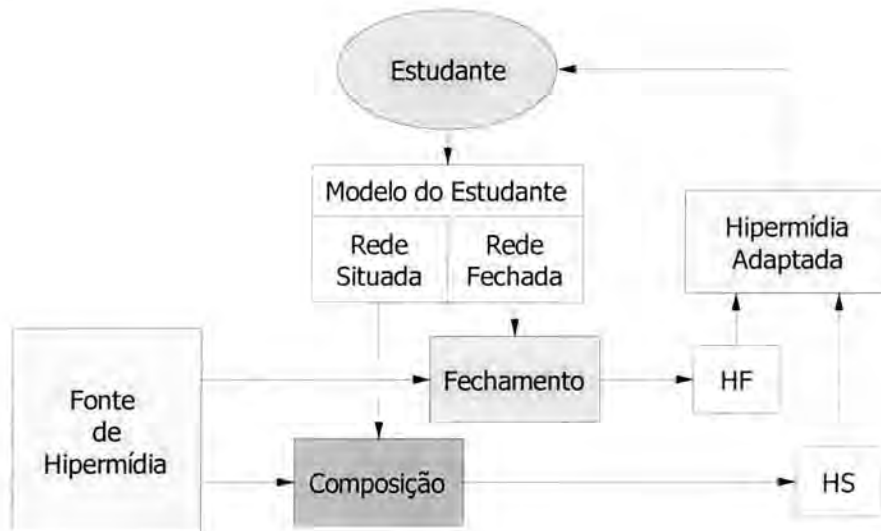


FIGURA 7.8 - Visão geral do processo de adaptação

A adaptação situada é obtida pela seleção, por meio de seus descritores, de documentos que exibam um grau de relevância G elevado em relação à rede situada presente no modelo do usuário. O grau de relevância é uma relação binária, anti-simétrica, calculada, como foi mostrado no capítulo 5, através do percentual de cobertura existente entre os conceitos que descrevem os objetos envolvidos.

A adaptação fechada é obtida pela aplicação das regras de transitividade e simetria sobre a frequência de ativação dos links na rede de hipermídia. As redes assim geradas são quantizadas pela determinação de um potencial de ativação P para cada link, sobre o qual a aplicação de um limiar de visibilidade e uma taxa de degradação Z garantem sua atualização.

Os dois modelos atuam como restrições na seleção da hipermídia disponibilizada pelo sistema e seus resultados combinados constituem a hipermídia adaptada que será apresentada ao estudante. Ao longo do processo o estudante poderá intervir muitas vezes, seja através de consultas, requisições, retornando feedback ou simplesmente navegando na rede hipermídia. A observação da sua ação é registrada para alimentar o processo de modelagem visto na seção 7.6.

Capítulo 8

Conclusões e Perspectivas de Trabalhos Futuros

8.1 Modelos Proativos para Hipermedia Adaptativa

O presente trabalho tem sua contribuição principal na área de modelos para hipermedia adaptativa. Ali foi possível, através da integração de duas representações complementares, uma simbólica e outra conexionista, obter considerável incremento na expressividade final obtida. Na visão adotada, a porção conexionista da representação captura os aspectos *dinâmicos* da navegação, enquanto que sua porção simbólica estabelece um relacionamento de *relevância situada* (isto é, dependente da situação considerada) entre páginas na web. O modelo resultante é centrado no usuário e evolui naturalmente a partir da interação deste com o sistema. Operando com duas perspectivas de representação, o sistema encontra-se em processo de permanente evolução, a qual decorre de sua interação com seus usuários. Estressando a representação em rede, esta idéia é apresentada de forma simplificada na figura 8.1.

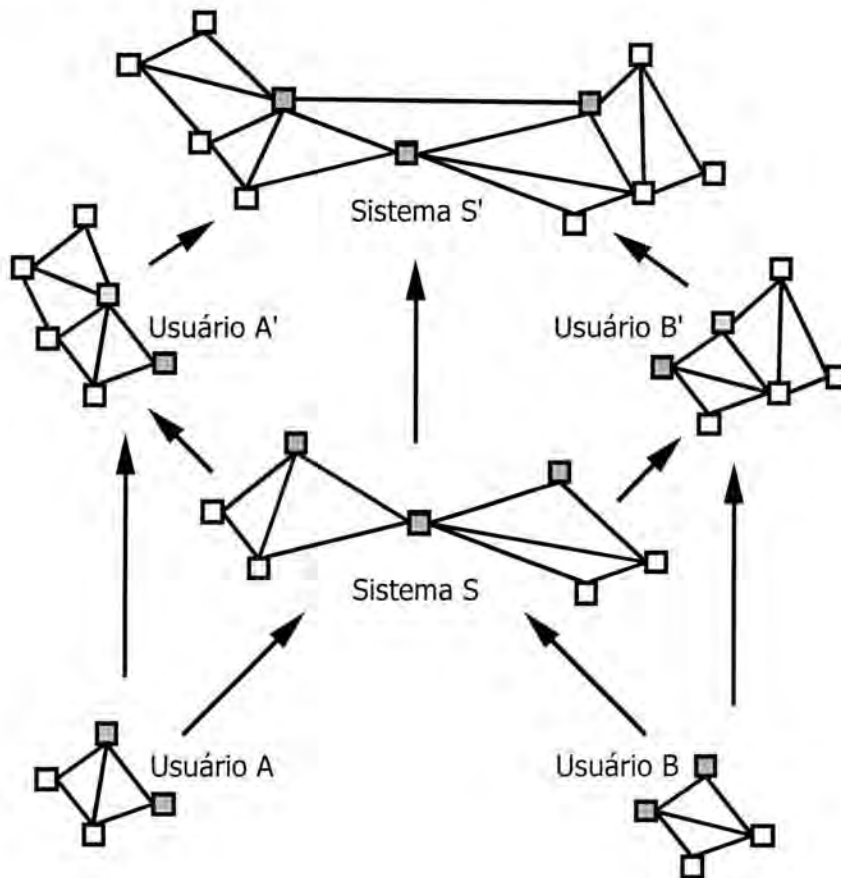


FIGURA 8.1: Evolução dos modelos de usuários e do sistema na representação em rede

Na figura 8.1 apresenta-se uma possível evolução da representação em rede de um sistema com dois usuários, de um estado S inicial para um estado seguinte S', onde a representação original dos usuários A e B também evolui para A' e B'. Pode-se observar que não só o estado seguinte do sistema depende da evolução do modelo do usuário como a recíproca também é verdadeira, isto é, a evolução dos modelos de usuários também ocorre na dependência da evolução do sistema. Isto estabelece um *comprometimento* entre os dois processos de evolução que decorrem da interação entre o usuário e o sistema. O sistema adapta-se ao usuário e este ao sistema, em um processo interativo contínuo que determina entre ambos uma permanente interdependência.

Entende-se então por adaptação *proativa* toda forma de adaptação produzida por algum mecanismo ou esquema capaz de prever possíveis estados futuros do ambiente e, se desejável, submetê-los diretamente aos usuários. Desta forma a adaptação proativa busca antecipar na dinâmica do sistema o surgimento de novos estados e configurações adaptando-os ao modelo do usuário. Este, por sua vez, também evolui no mesmo processo dinâmico, produzindo novas perspectivas de adaptação ao sistema. A aplicação de técnicas de adaptação proativa na web pode conduzir a bons resultados, uma vez que acelera a convergência em direção a resultados (ótimos locais) os quais de outra forma poderiam demorar mais tempo a ser atingidos, ou mesmo nem sequer ser considerados.

8.1.1 O Componente Conexionista

Mostrou-se que é possível capturar a evolução dinâmica do processo de navegação na web por meio de redes quantizadas onde cada link possui um *potencial de ativação* representando a probabilidade de que o link esteja ativado num próximo ciclo ou intervalo de tempo. Um algoritmo simples para a determinação de uma rede quantizada em função do potencial de ativação de seus links é o seguinte:

1. Iniciar com o potencial de ativação zerado para todos os links da rede considerada, em uma cópia de trabalho da rede do sistema;
2. Durante um certo intervalo de tempo, registrar a partir da navegação observada na rede a frequência com que cada link é percorrido ou ativado;
3. Sobre a rede assim obtida, quantizada com as frequências de ativação dos links aplicar as operações de fechamento simétrico e transitivo. Isto poderá fazer surgir novos links na rede com o respectivo potencial de ativação associado;
4. Sobre o potencial de ativação dos links da rede resultante do passo 3, aplicar um valor *limiar de visibilidade*, removendo, para efeitos práticos, os links cujos potenciais de ativação estejam abaixo do limiar;
5. Adicionar a rede obtida no passo 4 à representação *acumulada* do sistema, que modela sua evolução continuada ao longo de todos os ciclos;
6. Aplicar sobre a representação acumulada a taxa de degradação, que incide como um decremento percentual sobre os potenciais de ativação de todos os links da rede acumulada;
7. Repetir os passos 1 a 6, indefinidamente.

No algoritmo informalmente descrito acima, optou-se por aplicar as operações de fechamento simétrico e transitivo sobre a rede de trabalho, submetendo-a ao limiar de visibilidade e obtendo seu efeito localizado no ciclo antes de adicionar a rede resultante à representação acumulada do sistema. Outro efeito seria obtido caso essas operações fossem aplicadas diretamente sobre a rede acumulada ou sobre ambas as redes em diferentes momentos. Além disso, tendo em vista a escalabilidade natural dos modelos em rede, o mesmo algoritmo pode ser também aplicado a um único usuário, produzindo sua representação de base conexcionista. É importante observar entretanto que o componente conexcionista baseia-se exclusivamente nos links e na sua frequência de ativação, abstraindo totalmente os conteúdos dos nodos.

8.1.2 O Componente Simbólico

Complementando a visão conexcionista, o componente simbólico estabelece uma segunda rede entre os descritores de objetos hipermídia, agora baseada na *relevância* existente entre eles. Esta nova rede é construída de forma totalmente independente da rede dos potenciais de ativação anteriormente descrita. O objetivo aqui é estabelecer quão *relevante* é o conteúdo de um nodo para os demais. Trata-se portanto de uma rede sobre um relacionamento semântico. Esta rede possui associado ao seus links um *grau de relevância*, que é obtido na presença (situação) do modelo do usuário. A representação simbólica é muito mais estável do que a representação conexcionista e pode ser empregada como referência em casos de tomada de decisão.

A relação de relevância é determinada com base na teoria das situações, onde cada documento ou objeto hipermídia possui um descritor, que é uma *situação*. O descritor é formado por *ínfons*, que são informações atômicas que descrevem atributos do objeto considerado. Uma classe especial de ínfons são as *palavras-chave*, que possuem um *índice de importância*, em relação ao objeto descrito. Este índice permite classificar as palavras-chave no descritor de um objeto da rede, tornando mais precisa a representação. O *grau de relevância* é estabelecido entre dois objetos como o percentual de cobertura que a composição de seus descritores apresenta sobre cada um deles. O grau de relevância permite quantificar a rede e operar sobre ela com um limiar de visibilidade, a exemplo do que é feito no modelo conexcionista.

8.1.3 Integração dos Componentes

A integração dos modelos simbólicos e conexcionistas em uma única representação é possível, uma vez que ambos são modelados como redes sobre os mesmos nodos, como é mostrado na figura 5.4. Fica evidenciado que o modelo resultante desta integração possui ampla expressividade, permitindo representar aspectos estáticos, dinâmicos e organizacionais dos objetos modelados. O emprego deste método de modelagem integrada permite assim representar um processo de adaptação *proativa*, onde novas possibilidades são introduzidas em *antecipação* aos objetivos do usuário.

Concebido para aplicação na web, há um mapeamento direto entre os conceitos do modelo proposto e os que a fundamentam. Assim foi possível projetar um mecanismo inferencial para gerar novos espaços de navegação na rede em conformidade com os modelos de seus usuários. Este mecanismo atua suportado pelos dois componentes de modelagem, cuja integração possibilita traduzir, em um único *framework*, visões complementares do mesmo objeto hipermídia.

8.2 Uma Arquitetura para HA Proativa

No capítulo 6 foi apresentada a proposta de uma arquitetura genérica para sistemas de HA proativa na web. A arquitetura foi concebida a partir da interação entre três processos assíncronos e até certo ponto independentes: o processo de *modelagem*, o processo de *adaptação* e o processo de *interfaceamento*. O primeiro executa a criação e manutenção de modelos de usuários na *base de modelos de usuários* (bmu). O segundo utiliza os modelos de usuários na bmu para adaptar consultas e também a hipermídia recebida em resposta. Finalmente o processo de interfaceamento oferece suporte à navegação e à apresentação da hipermídia ao usuário. Na figura 6.2 esta arquitetura é apresentada, destacando cada um dos três processos citados.

É interessante notar que o tipo de modelagem proposto irá apresentar seus melhores resultados após uma interação prolongada com o sistema, possibilitando o refinamento de seu modelo e em consequência uma maior precisão de resposta. Esta característica conduziu a uma arquitetura adaptativa em que os processos envolvidos ocorrem concorrentemente, sendo sincronizados por meio de *ciclos* de modelagem e adaptação. A concorrência entre os processos favoreceu o projeto de uma arquitetura orientada a agentes.

Os algoritmos associados a cada um dos componentes da arquitetura foram desenvolvidos perseguindo o objetivo de manter a arquitetura *genérica*, isto é, configurável de acordo com *esquemas* de modelagem e adaptação. O uso de tais esquemas confere ao sistema um elevado grau de flexibilidade, permitindo-lhe adotar diferentes configurações para atender a propósitos reais ou experimentais. A arquitetura foi especificada em lógica, empregando Prolog como linguagem de especificação.

8.3 ia@net: Adaptação Proativa na Web

No endereço <http://ha.ucpel.tche.br/> encontra-se a página do projeto ia@net, um sistema educacional experimental online de Inteligência Artificial projetado para suportar adaptação proativa. O sistema foi construído em html dinâmico com scripts AWK na interface e Prolog na programação dos processos de adaptação e modelagem. Em ambos os casos os modelos são representados por redes conceituais quantizadas. O modelo do usuário aqui é o próprio modelo do aluno, que evolui tornando-se mais preciso ao longo de sua interação com o sistema.

As informações para o processo de modelagem são obtidas a partir do rastreamento das atividades do aluno e também através de feedback direto que este é incentivado a fornecer. As páginas do sistema são apresentada em estruturas de frames com uma parte estática, com títulos, recursos e outros elementos comuns do sistema, e uma parte dinâmica com o conteúdo hipermídia especificamente adaptado ao modelo do estudante.

O sistema reúne um grande número de recursos na área de inteligência artificial, incluindo artigos, textos, programas, links, etc. Diferentes configurações dos parâmetros de modelagem e adaptação nos respectivos esquemas permite projetar sobre o sistemas estruturas de estudo e pesquisa para atender diferentes objetivos de aprendizado, tais como cursos de extensão (em profundidade variável), disciplinas de graduação, disciplinas de pós-graduação, programas de estudo individual, treinamento específico, pesquisas, etc.

Sistemas educacionais online podem empregar vantajosamente métodos de adaptação proativa para a pesquisa e seleção de material relevante e de interesse direto do estudante em seus objetivos de aprendizado. O método da adaptação proposto, integrando a lógica das situações a um modelo conexionista, permite a contínua evolução do modelo do estudante. O sistema experimental ia@net representa um esforço inicial de implementação dessa proposta, cujos resultados poderão indicar os rumos de trabalhos e desenvolvimentos futuros.

8.4 Perspectivas de Trabalhos Futuros

As aplicações de HA caracterizam-se por manter *modelos* de seus usuário, aos quais os objetos apresentados e a navegação entre eles devem ser adaptados. Uma área de aplicação em plena expansão é a dos sistemas educacionais online. Aqui um primeiro resultado concreto da pesquisa realizada é a disciplina de Inteligência Artificial (código 70021) que será oferecida totalmente online aos estudantes dos cursos de bacharelado em Ciência da Computação e Análise de Sistemas da Universidade Católica de Pelotas (UCPel) a partir do primeiro semestre de 2000. A disciplina será ministrada empregando os recursos do sistema ia@net, incluindo os componentes de modelagem e adaptação. Sua oferta constitui um teste prático fundamental para o sistema, onde uma as técnicas de hipermídia adaptativa serão empregada em associação com a plataforma didático-pedagógica oferecida pela web. Espera-se que a experiência adquirida com o desenvolvimento da disciplina possa permitir o desenvolvimento de uma metodologia para iniciativas semelhantes.

Outro campo de aplicação que necessita modelagem do usuário e que apresenta amplas perspectivas de expansão nos próximos anos é o dos assistentes pessoais inteligentes, capazes de atuar simultaneamente na web e no desktop do usuário executando tarefas de classificação, seleção, agenda, pesquisa na web, filtragem dos serviços de notícias e correio eletrônico, organização de informações e assim por diante. Todas essas atividades podem beneficiar-se do uso de um modelo do usuário, de acordo com o qual as atividades do assistente seriam executadas. Um projeto de pesquisa elaborado pelo Grupo de Pesquisa em Inteligência Artificial da UCPel orientado ao desenvolvimento de assistentes pessoais inteligentes para a web está planejado para iniciar em março de 2000, a partir dos modelos proativos aqui apresentados.

Uma terceira perspectiva de continuação do trabalho desenvolvido na presente tese envolve investigar a modelagem de grupos e a interação direta entre usuários ou entre usuários e grupos. Dada a escalabilidade dos modelos em rede, é fácil obter o modelo de um grupo a partir dos modelos de seus componentes. Modelos de grupos podem ser construídos para diversas finalidades, segundo diferentes classificações de possíveis componentes. A base de modelos de usuários construída na implementação do sistema ia@net pode ser empregada para modelar grupos de usuários classificados segundo suas características semelhantes. Esta forma de modelagem encontra aplicação em organizações, associações e qualquer classe de usuário. O objetivo é representar em um único modelo o resultado da composição dos modelos dos componentes do grupo. O modelo resultante pode ser empregado de diversas formas, traduzindo sempre as características comuns a todos os (ou a maioria dos) usuários do grupo.

Anexo 1

Uma especificação para a adaptação de requisições do usuário em Prolog

```

%-----
%
% Módulo adaptaReq - Adaptação da requisição do usuário antes de
% submetê-la ao servidor hipermídia.
%
%-----
%
% adaptaReq/4 - Adapta a requisição formulada pelo usuário ao
% seu modelo, produzindo uma nova requisição para ser enviada ao
% servidor Web.
%
% A chamada é:
%
%     adaptaReq(Req, Id, EAREq, NovaReq),
%
% onde:
%
%   Req:   Requisição original do usuário, obtida através de
%          um form gerado na página apresentada ao usuário.
%
%   Id:    Identificação do usuário. Aponta para um conjunto de
%          sentenças lógicas que descrevem o modelo do usuário.
%
%   EAREq: Esquema de adaptação utilizado, representado por um
%          conjunto de sentenças lógicas.
%
%   NovaReq: Nova requisição a ser submetida ao servidor hipermídia,
%            inferida a partir da requisição original e do
%            modelo do usuário, através do esquema de adaptação
%            selecionado.
%-----

adaptaReq(Req,Id,EAREq,NovaReq):-          % Nível mais alto do
    req(Req,Id),                          % procedimento de
    eareq(EAREq,Id),                      % adaptação da requisição
    infere(Req,Id,EAREq,NovaReq).         % do usuário.

%-----
%
% As requisições originais de cada usuário são representadas como
% fatos Prolog req/2:
%
%   req(Req, Id), onde:
%
%   Req:   É uma lista das requisições de um usuário que podem
%          ser urls, palavras-chaves, recursos, etc.
%
%   Id:    Identificação do usuário requisitante.
%-----

```

```

%-----
%
% A seguir exemplos de possíveis fatos req/2, onde o usuário pode ser
% um grupo de pesquisa(gpia), um indivíduo (lpalazzo) ou uma entidade
% não especificada (anon), cada um destes usuários possui um modelo
% apropriado dado por um conjunto de sentenças lógicas.
%-----
req(['http://www.amzi.com/evaldist.zip'], gpia).
req(['adaptive hypermedia',tutorial,primer, lpalazzo).
req(['http://um.org','http://ia.ucpel.tche.br/index.html'], anon).

%-----
%
% O modelo adotado pressupõe que esquemas de adaptação específicos
% (representados por conjuntos de sentenças lógicas) possam ser
% carregados para participar da inferência da nova requisição (que
% é o objetivo deste procedimento. Tal esquema deve estar em uma
% base de dados, sendo recuperado pelo procedimento eareq/2:
%
%   eareq(EAReq, Id, EA), onde
%
%   EAReq: Chave do esquema de adaptação, que será recuperado
%           como um conjunto de sentenças lógicas de uma base
%           de esquemas de adaptação.
%
%   Id:    Identificação do usuário que usa o correspondente
%           esquema de adaptação.
%
%   EA:    Lista de sentenças lógicas descrevendo o esquema
%           de adaptação empregado pelo usuário.
%-----

eareq(EAReq,Id,EA):-                % Introduce no programa um módulo
    carrega(EAReq,Id,EA).          % descrevendo o esquema de
                                   % adaptação (EA) para o usuário.
carrega(_,_,[ ]).                  % EA é uma lista de sentenças
carrega(EAReq,Id,[X|Y]):-          % lógicas', é introduzido no
    valida(EAReq,Id,X),             % programa por assert/1
    assert(X),                       % recursivamente aplicado sobre
    carrega(EAReq,Id,Y).            % os componentes da lista EA.

%-----
%
% O último procedimento é a "máquina de inferência" que produz uma nova
% requisição para ser submetida ao servidor web a partir da requisição
% original, do modelo do usuário e do esquema de adaptação considerado.
% Para isto se produz uma situação pela composição de Req, MU e EA e no
% contexto desta situação se infere indutivamente novaReq. O
% procedimento é infere/4:
%
%   infere(Req,Id,EAReq,NovaReq), onde:
%

```

```

%      Req:      É a requisição original do usuário, conforme
%              vista anteriormente.
%
%      Id:       É a identificação do usuário, a partir da qual
%              seu modelo vai ser recuperado e inserido na
%              situação considerada.
%
%      EAREq:    Identifica um particular esquema de adaptação a
%              ser empregado nesta inferência.
%
%      NovaReq:  É a nova requisição produzida adaptativamente.
%-----

```

```

infere(Req,Id,EAREq,NovaReq):-
    situação(Req,Id,EAREq,Situação),
    demo(Situação,NovaReq).

```

```

situação(Req,Id,EAREq,Situação):-
    modelo(Id,UM),
    eareq(EAREq,Id,EA),
    monta(Req,UM,EA,Situação).

```

```

%-----
%
% O predicado demo/2 modela as regras de inferência da lógica das
% situações, apresentada no capítulo 5, para produzir NovaReq, levando em
% conta a requisição original, mas também o modelo do usuário que a emitiu.
%-----

```

Anexo 2

Página de recursos para o
Módulo Um do ia@net.

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

Recursos de IA Online - Módulo Um

Atualizado em 12/10/1999

[Revistas e Jornais](#)
[Referências](#)
[Livros, Autores e Bibliografias](#)
[Organizações e Conferências](#)
[Grupos de Pesquisa](#)
[Empresas](#)
[Cursos de IA Online](#)
[Links Diversos](#)

Revistas e Jornais

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AI Magazine](#)
[AI Review](#)
[Artificial Intelligence*](#)
[CS Tech Reports](#)
[Computational Intelligence](#)
[ETAI Journal](#)
[IEEE Expert](#)
[IIT list](#)
[JAIR*](#)
[JETAI](#)
[Machine Learning](#)
[PC AI](#)

Referências

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[Dictionary of Computing](#)
[AI Patents](#)
[AIMA Bibliography*](#)
[AIMA Table of Contents](#)
[AIMA Home*](#)
[Intro to AI](#)
[Promises of Computing](#)
[What is AI*](#)
[AI Education Repository](#)
[AI FAQ*](#)
[AI Virtual Library](#)
[AI on the WWW](#)
[CMU list](#)
[Canadian IIT*](#)
[Excite's list](#)
[GSB/GSL \(MIT\)](#)
[Generation 5](#)
[London Collection](#)
[Meta-search list](#)
[Outsider's Guide](#)
[SIGART Other](#)
[SIGART Subject List](#)
[Yahoo list](#)
[World Lecture Hall](#)

Livros, Autores e Bibliografias

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AIMA Bibliography*](#)
[Amazon.com's list*](#)
[Bender](#)
[Comparison](#)
[Dean](#)
[Encyclopedia of AI*](#)
[Ginsberg](#)
[Karlsruhe Bibliography](#)
[Kurzweil](#)
[Manitoba Bibliography](#)
[Rich & Knight](#)
[Russell & Norvig*](#)
[Russell & Norvig Book in pt form](#)
[Saarbrucken](#)
[Vienna Bibliography](#)
[Winston](#)

Organizações e Conferências

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AAAI](#)
[ACM list](#)
[AISB](#)
[IEEE](#)
[SIGART*](#)
[AAAI-97](#)
[ACM list](#)
[IEEE list](#)
[IJCAI](#)
[International list](#)
[Neural Net Conferences](#)

Grupos de Pesquisa

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AIAI](#)
[Austrian AI](#)
[Brown](#)
[CMU*](#)
[Chicago](#)
[DFKI Saarbrucken](#)
[EKSL \(UMass\)](#)
[Edinburgh](#)
[Georgia Tech](#)
[IIT list](#)
[Illinois](#)
[Irvine](#)
[MIT*](#)
[Michigan](#)
[New S. Wales](#)
[Pittsburgh](#)
[Rochester](#)
[Santa Fe](#)
[Stanford*](#)
[Texas](#)
[UPenn](#)
[USC/ISI*](#)
[Washington](#)
[Xerox PARC](#)
[Yale](#)

Empresas

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AI Companies*](#)
[Acknosoft](#)
[Ascent](#)
[Blackboard](#)
[Brightware](#)
[Carnegie Group](#)
[Computational Logic](#)
[Cycorp](#)
[DataMind](#)
[Excalibur](#)
[ExperTelligence](#)
[Gensym](#)
[Haley Enterprise](#)
[Harlequin](#)
[Hugin](#)
[IIT list*](#)
[Inference](#)
[Intellicorp](#)
[Knowledge Industries](#)
[Kurzweil](#)
[LPA](#)
[Level 5](#)
[Lumina](#)
[NetBot](#)
[Neuraltech](#)
[Neuron Data](#)
[Teknowledge](#)
[Thinking Machines](#)
[Ultragem](#)
[Yahoo list](#)

Cursos de IA Online

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[Principles of AI \(Iowa\)](#)
[AI Tutorials](#)
[Intelligent Interactive Media \(MIT\)](#)
[AI \(MIT\) pub](#)
[AI Online Courses/Lectures](#)
[AI Methods \(Nottingham\)](#)
[Introduction to AI \(Leeds\)](#)
[Introduction to AI \(Toronto\)](#)
[AI \(Purdue\)](#)
[AI \(Southern California\)](#)
[Introduction to AI and Expert Systems](#)
[Introduction to AI \(Carnegie Melon\)](#)
[Introduction to AI \(BYU\)](#)
[Introduction to AI \(FIT\)](#)
[An Introduction to AI \(UCT\)](#)
[AI \(NCST\)](#)
[Genetic Machine Learning](#)
[Introduction to Neural Networks \(UVA\)](#)
[Machine Learning](#)
[Introduction to Neural Networks](#)
[Introduction to AI \(NUS\)](#)
[Queen Mary & Westfield College](#)
[AI \(Massey\) Intro to AI \(UBC\)](#)
[AI \(Harvard\)](#)
[AI \(UNSW\)](#)
[Databases and AI 3 \(HWU\)](#)
[AI \(UWM\)](#)

[Intro to AI \(Gatech\)](#)
[AI \(WPI\)](#)
[AI \(Texas\)](#)
[AI \(VILL\)](#)
[AI \(Berkeley\)](#)
[AI \(MIT\)](#)
[AI \(UFL\)](#)
[The Intro to AI Show \(RPI\)](#)
[AI \(JHU\)](#)
[AI \(UARK\)](#)
[AI \(Nottingham\)](#)
[Knowledge Systems Engineering](#)
[AI1, AI2 \(Cardiff\)](#)

Links Diversos

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

[AIMA \(Official Website\)](#)
[Intelligent Systems Lab \(NUS\)](#)
[Knowledge Based Engineering \(JKU\)](#)
[Intelligent Internet Agents](#)
[Innovative Applications of AI](#)
[Decision Analysis and Expert Systems](#)
[Characteristics of Expert Systems](#)
[Interesting AI Demos and Projects](#)
[AI on the Web](#)
[Neural Network Applications \(JSAIC\)](#)
[AI Tutorials](#)
[AI Courses](#)
[Useful AI-Related Pages](#)
[Intelligent search engines](#)
[AI Should we? And then how?](#)
[AI related FAQs 1 | 2](#)
[Alan Turing Home Page](#)
[Kasparov vs Deep Blue](#)
[Journal of AI Research](#)
[AI Materials \(Kingston\)](#)
[Kingston University](#)
[Java Demo Applets for Artificial Intelligence](#)
[Artificial Intelligence \(CSM\)](#)
[AI Lecture Notes \(McGill University\)](#)
[Institute of IT \(AI Resources\)](#)
[Journal of AI Research](#)
[AI: Modern Approach \(Source Codes\)](#)
[AI \(Sunderland\)](#)
[AI \(Projects and Solutions\)](#)
[Software for "Programming AI: Methods, Tools and Apps" Book 2](#)
[Smart Moves: Intelligent Pathfinding](#)
[AI/Game Programming Resources](#)
[Game Programming with AI Games Articles](#)
[Loebner Prize "The First Turing Test"](#)
[CLIPS: A Tool for Building ES](#)
[The Age of Robots](#)
[What is Artificial Intelligence?](#)
[Generation 5: AI Repository](#)
[AI Timeline \(até 1994\)](#)
[The History of AI \(até 1991\)](#)
[The History of AI](#)

© 1999

← ↑ → COMP IA 1 2 3 4 5 6 7 8 9 10 INFO

Bibliografia

- [ARM 95] ARMSTRONG, Robert et al. **WebWatcher: A learning apprentice for the World Wide Web**. Disponível por WWW em <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-6/web-agent/www/webagent-plus/webagent-plus.html> (5 out. 1998).
- [ASH 56] ASHBY, R. **Introduction to Cybernetics**. London: Methuen, 1956.
- [BAR 89] BARWISE, J. **The Situation in Logic**. Stanford: CSLI Publications, 1989. 337p.
- [BAR 90] BARWISE, J.; ETCEMENDY, J. Information, infons, and inference. **Situation Theory and its Applications**, Stanford, v.1, n.26, p.33-78, 1990.
- [BEA 98] BEAUMONT, Jan H. User modeling in the interactive tutoring system ANATOM-TUTOR. In: BRUSILOVSKY, Peter et al. **Adaptive Hypertext and Hypermedia**. Dordrecht: Kluwer Academic Publishers, 1998. p.91-115.
- [BER 94] BERNERS-LEE, Tim; MASINTER, L.; McCAHILL, M. **Uniform Resource Locators (URL), RFC 1738, CERN**. Minnesota: Xerox PARC, 1994.
- [BOL 96] BOLLEN, J.; HEYLIGHEN, F. **Algorithms for the Self-Organization of Distributed Multi-user Networks**. Disponível por WWW em <http://pespmc1.vub.ac.be/papers/SelfOrganWWW.html> (19 fev 1998).
- [BOY 94] BOYLE, C.; ENCARNACIÓN, A. O. MetaDoc: an adaptive hypertext reading system. **User Modeling and User-Adapted Interaction**, Dordrecht, v.4, n.1, p.1-19, 1994.
- [BRP 94] BRUZA, P. D.; HUIBER T. W. C. Investigating aboutness axioms using information fields. In: ANNUAL INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 17., 1994, Dublin. **Proceedings...** Berlin: Springer-Verlag, 1994. p.112-121.
- [BRU 92] BRUSILOVSKY, Peter. A framework for intelligent knowledge sequencing and task sequencing. In: FRASSON, C.; GAUTHIER, G.; MCCALLA, G. I. **Intelligence Tutoring Systems**. Berlin: Springer-Verlag, 1992. p.499-506.
- [BRU 93] BRUSILOVSKY, Peter; ZYRIANOV, M. Intelligence tutor, environment and manual for physical geography. In: INTERNATIONAL PEG CONFERENCE, 7., 1993, Edinburgh. **Proceedings...** Berlin: Springer-Verlag, 1993.
- [BRU 94] BRUSILOVSKY, Peter; PESIN, L. ISIS-Tutor: An adaptive hypertext learning environment for CDS/ISIS users. Disponível por WWW em http://cs.joensuu.fi/~mtuki/www_clce.270296/Brusilov.html (28 maio 1997).
- [BRU 96] BRUSILOVSKY, Peter. Methods and Techniques of Adaptive Hypermedia. **User Modeling and User-Adapted Interaction. Special issue on adaptive hypertext and hypermedia**, Dordrecht, v.6, n.2-3, p.87-129, 1996.

- [BRU 97] BRUSILOVSKY, Peter. Efficient techniques for Adaptive Hypermedia. In: NICHOLAS, C.; MAYFIELD, J. (Eds.). **Intelligent Hypertext: Advanced techniques for the World Wide Web**. Berlin: Springer-Verlag, 1997. p. 12-30. (Lecture Notes in Computer Science, v.1326).
- [BRU 98] BRUSILOVSKY, Peter. A Study of User Model Based Link Annotation in Educational Hypermedia. **Journal of Universal Computer Science**, Berlin, v.4, n.4, p.428-448, Apr. 1998.
- [BUS 45] BUSH, Vanevar. As we may think. **The Atlantic Monthly**, 1945. Disponível por WWW em <http://www.isg.sfu.ca/~duchier/misc/vbush> (25 mar. 1998)
- [CAM 88] CAMPBELL, Brad; GOODMAN, Joseph. HAM: A general purpose hypertext abstract machine. **Communications of the ACM**, New York, v.31, n.7, p.856-861, July 1988.
- [CHE 92] CHEVALLET, J-P. Un modèle logique de recherche d'informations appliqué au formalisme des graphes conceptuels. In: **Le prototype ELEN et son expérimentation sur un corpus de composants logiciels**. PhD thesis, Université Joseph Fourier, Grenoble, 1992. Disponível por ftp em <ftp://ftp.imag.fr/pub/SIRI/publications/1989-1992/TheseJPC> (17 jul. 1998).
- [COO 71] COOPER, W.S. **A definition of relevance for information retrieval**. Disponível por WWW em <http://slis6000.slis.uwo.ca/601/test/documents/doc0445.html> (12 ago. 1998).
- [DeB 92] De BRA, Paul et al. **An Extensible Data Model for Hyperdocuments**. Disponível por WWW em <http://www.win.tue.nl/~debra/echt92/final.ps> (18 fev. 1998).
- [DeB 97] De BRA, Paul; CALVI, Licia. Creating adaptative hyperdocuments for and on the web. In: WEBNET – WORLD CONFERENCE OF THE WWW, INTERNET AND INTRANET. Charlottesville, 1997. **Proceedings...** Charlottesville: Association for the Advancement of Computing Education, 1997.
- [DeR 93] De ROSIS, F.; De CAROLIS, B.; PIZZUTILO, S. **User tailored hypermedia explanations**. Disponível por WWW em <http://www.cs.bgsu.edu/hypertext/adaptive/deRosis.html> (17 maio 1998).
- [DEV 91] DEVLIN, K. **Logic and Information**. Cambridge: Cambridge University Press, 1991.
- [DRE 81] DRETSKE, F.I. **Knowledge and the Flow of Information**. Cambridge: Basic Blackwell Publisher, 1981.
- [FIS 90] FISCHER, G. et al. Minimalist explanations in knowledge-based systems. ANNUAL HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 23., 1990, Kailua-Kona. **Proceedings...** Los Alamitos: IEEE Computer Society Press. p. 309-307.
- [GON 95] GONSCHOREK, M.; HERZOG, C. Using hypertext for an adaptative helpsystem in an intelligent tutoring system. In: AI-ED, WORLD CONFERENCE ON ARTIFICIAL INTELLIGENCE IN EDUCATION, 7., 1995, Washington. **Proceedings...** Charlottesville: AACE, 1995. p. 274-281.

- [GRU 93] GRUNST, G. Adaptive hypermedia for support systems. In: SCHNEIDER-HUFSCHMIDT, M.; KÜHME, T.; MALINOWSKI, U (Eds.). **Adaptive user interfaces: Principles and practice**. Amsterdam: North-Holland, 1993. p. 269-283.
- [HAD 95] HADDAD, H.: **VIIP: An iconic-indexing approach for video**. Disponível por WWW em <http://www.cgi.polimit.it/viip.htm> (27 abr. 1999).
- [HAL 90] HALASZ, Franz; SCHWARTZ, Mayer. The Dexter hypertext reference model. In: **HYPERTEXT STANDARDIZATION WORKSHOP, 1990**, Gaithersburg. **Proceedings...** Gaithersburg: NIST Special Publication, 1990. p. 150-178
- [HEY 88] HEYLIGHEN, F. **Building a Science of Complexity**. Disponível por WWW em <http://pespmc1.vub.ac.be/papers/BuildingComplexity.html> (31 jul. 1998).
- [HEY 96] HEYLIGHEN, F.; BOLLEN, J. The World Wide Web as a Superbrain: From Metaphor to Model. In: **CYBERNETICS AND SYSTEMS, 1996**, Singapore. **Proceedings...** Singapore: World Science, 1996. p. 78-92.
- [HOH 96] HOHL, H. et al. Hypadapter: An adaptive hypertext system for exploratory learning and programming. **User Modeling and User-Adapted Interaction. Special issue on adaptive hypertext and hypermedia**, Dordrecht, v. 6, n.2-3, p.131-156, 1996.
- [HÖÖ 98] HÖÖK, K. et al. A glass box approach to adaptive hypermedia. In: BRUSILOVSKY, Peter et al. **Adaptive Hypertext and Hypermedia**. Dordrecht: Kluwer Academic Publishers, 1998. p. 143-170.
- [HUG 84] HUGHES, G.E.; CRESSWELL, M.J. **A Companion to Modal Logic**. London: Methuen, 1984.
- [HUI 94] HUIBERS, T.W.C.; BRUZA, P.D. **Situations, a General Framework for Studying Information Retrieval**. The Netherlands: Department of Computer Science, Utrecht University, 1994. (Technical Report UU-CS-1994-15).
- [HUI 95] HUIBERS, T.W.C.; OUNIS, I.; CHEVALLET, J-P. **Axiomatization of a conceptual graph formalism for information retrieval in a situated framework**. France: The Laboratory of Génie Informatique, Groupe MRIM of Grenoble, 1995. (Technical Report RAP95-004).
- [KAP 93] KAPLAN, Craig; FENWICK, J.; CHEN, J. Adapted hypertext navigation based on user goals and context. **User Modeling and User-Adapted Interaction**, Dordrecht, v.5, n.3, p. 193-220, 1993.
- [KAY 94] KAY, J.; KUMMERFELD, R. **Adaptive hypertext for individualized instruction**. Disponível por WWW em <http://www.cs.bgsu.edu/hypertext/adptative/Kay.html> (7 out. 1998).
- [KAU 93] KAUFFMAN, S. A. **Origins of Order: self-organization and selection in evolution**. New York: Oxford University Press, 1993.
- [KAU 95] KAUFFMAN, Stuart A. **At Home in the Universe**. New York: Oxford University Press, 1995.

- [KOB 94] KOBSA, A. et al. KN-AHS: An adaptive hypertext client of the user modeling system BGP-MS. In: INTERNATIONAL CONFERENCE ON USER MODELING, 4., 1994, Hyannis. **Proceedings...** Hyannis: The MITRE Corporation, 1994. p. 31-36
- [LAL 93] LALMAS, M.; VAN RIJSBERGEN, C. J. A model of an information retrieval system based on situation theory and dempster-shafer theory of evidence. In: WORKSHOP ON INCOMPLETENESS AND UNCERTAINTY IN INFORMATION SYSTEMS, 1993. **Proceedings...** Montreal: Concordia University, 1993. p 62-67.
- [LAN 86] LANDMAN, F. **Towards a Theory of Information**. Dordrecht: Foris, 1986.
- [LAN 90] LANGE, Danny B. A formal model of hypertext . In: MOLINE, Judi et al. In: HYPERTEXT STANDARDIZATION WORKSHOP, 1990, Gaithersburg. **Proceedings...** Gaithersburg: NIST Special Publication, v. 500-178, 1990. MD 20899.
- [LIN 95] LINARD, M.; ZEILIGER, L. Designing a navigational support for educational software. In: BLUMENTAL, B. et al. **Human-Computer Interaction**. Berlin: Springer-Verlag, 1995. (Lecture Notes in Computer Science, v. 1015).
- [LOK 98] LOKE, Seng W. **Adding Logic Programming Behaviour to the World Wide Web**. Melbourne: University of Melbourne Press, 1998.
- [MAR 77] MARON, M.E. On indexing, retrieval and the meaning of about. **Journal of the American Society for Information Science**, New York, n.28, p.38-43, Jan. 1977.
- [MAT 96] MATHÉ, N.; CHEN, J. User-centered indexing for adaptive information access. **User Modeling and User-Adapted Interaction, Special issue on adaptive hypertext and hypermedia**, Dordrecht, v.6, n.2-3, 1996.
- [MAY 95] MAYER-KRESS, G.; BARCZYS, C. The Global Brain as an Emergent Structure from the Worldwide Computing Network. **The Information Society**, Bloomington, v.11, n.1, p. 1-27 1995
- [MIC 96] MICARELLI, A.; SCIARONE, F. A case based toolbox for guided hypermedia navigation. In: UM, INTERNATIONAL CONFERENCE ON USER MODELING, 5., 1996, Kailua-Kona. **Proceedings...**Kailua-Kona: User Modelling Inc., 1996. p.129-136.
- [MUK 95] MUKHERJEA, S.; FOLEY, J. D. **Visualising the World Wide Web with a navigational view bulde**. Disponível por WWW em <http://www.igd.fhg.de/www/www95/proceedings/papers/44/mukh/mukh.html> (14 set. 1997)
- [NIE 90] NIE, J. **Un modèle logique général pour les systèmes de recherche d'informations** - Application au prototype RIME. PhD thesis, Universit'e Joseph Fourier, Grenoble, 1990. Disponível por WWW em http://www-eiscat.ujf-grenoble.fr/index_fr.html (4 jul. 1997).
- [NUA 99] NUA Surveys. Disponível por WWW em <http://www.nua.ie/surveys> (10 dez 1999).

- [PAL 95] PALAZZO, L. A. M. Evolução das Técnicas de Análise e Modelagem de Sistema de Informações. **Boletim Eletrônico no.1 da Associação dos Bacharéis em Informática do Rio Grande do Sul (ABINFO - RS)**, Porto Alegre, ago. 1995. 12p.
- [PAL 96a] PALAZZO, L. A. M.; CASTILHO, J. M. V. Sistemas de Informações Inteligentes: Uma Perspectiva Cibernética. **Cadernos de Informática e Educação**, Pelotas, n.1, v.1, p.53-71, 1996.
- [PAL 96b] PALAZZO, L. A. M. **Aspectos da Modelagem de Sistemas de informações Inteligentes**. Porto Alegre: CPGCC da UFRGS, ago. 1996. 97p. Exame de Qualificação.
- [PAL 97] PALAZZO, L. A. M. (Org). OFICINA DE INTELIGÊNCIA ARTIFICIAL – CONCEITOS, 1., 1997, Pelotas. UCPEL/ESIN. **Anais...** Pelotas: Educat , 1997. 107p.
- [PAL 98a] PALAZZO, L. A. M.; CASTILHO, J. M. V. Modelagem de Redes em Sistemas de Informação Geográficas. In: OFICINA DE MULTIMÍDIA E GEOPROCESSAMENTO, 1., 1998, Pelotas. **Anais...** Pelotas: Educat, 1998. p.31-42.
- [PAL 98b] PALAZZO, L. A. M.; PORTO, P. R. P. (Orgs). In: OFICINA DE INTELIGÊNCIA ARTIFICIAL, 2., 1998, Pelotas. UCPEL/ESIN. **Anais...** Pelotas: Educat, 1998.
- [PAL 98c] PALAZZO, L. A. M.; CASTILHO, J. M. V. Processamento Semântico de Informações Situadas. In: OFICINA DE INTELIGÊNCIA ARTIFICIAL. UCPEL/ESIN, 2., 1998, Pelotas. **Anais...** Pelotas: Educat. p.111-125.
- [PAL 98d] PALAZZO, L. A. M.; CASTILHO, J. M. V.; COSTA, A. C da R. Self-organization of Information Networks. In: SEMANA ACADÊMICA DO CPGCC DA UFRGS, 3., 1998, Porto Alegre. **Anais...** Porto Alegre: CPGCC da UFRGS, 1998.
- [PAL 98e] PALAZZO, L. A. M.; COSTA, A. C. da R. **Modelos Ativos para Hipermídia Adaptativa**. Pelotas: ESIN, 1998. 18p.
- [PÉR 95] PÉREZ, T.; GUTIÉRREZ, J.; LOPISTÉGUY, P. An adaptive hypermedia system. In: AI-ED, WORLD CONFERENCE ON ARTIFICIAL INTELLIGENCE IN EDUCATION, 7., 1995, Washington. **Proceedings...** Charlottesville: AACE, 1995. p. 351-358.
- [PRI 84] PRIGOGINE, I.; STENGERS, I. **Order Out Of Chaos**. New York: Bantam Books, 1984.
- [RIC 91] RICH, Elaine; KNIGHT, Kevin. **Artificial Intelligence. 2. ed**. New York: McGraw-Hill, 1991.
- [RIJ 93] RIJSBERGEN, C. J. van. **Two essays in information retrieval**. Glasgow: University of Glasgow Press, 1993. (Departmental Research Report IR-93-3).
- [RIV 94] RIVLIN, Ehud et al. Navigating in hyperspace: Designing a structure-based toolbox. **Communications of the ACM**, New York, v.37, n.2 , p.87-96, 1994.
- [RUS 95] RUSSEL, Stuart; NORVIG, Peter. **Artificial Intelligence: A Modern Approach**. New Jersey: Prattice-Hall Inc, 1995.

- [SCH 89] SCHNEIDERMAN, B.; KEARSLEY, G. **Hypertext Hands-On! An Introduction to a New Way of Organizing and Accessing Information.** München: Addison-Wesley, 1989.
- [SCH 96] SCHWARZ et al. **World-wide intelligent textbooks.** Disponível por WWW em <http://www.contrib.andrew.cmu.edu/~plb/ED-MEDIA-96.html> (6 jun. 1999).
- [SHA 56] SHANNON, C. E. **Bandwagon.** IRE Transactions on Information Theory, New York, v.2, n.1 p.2-3, 1956.
- [SHA 64] SHANNON, C. E.; WEAVER, V. **Mathematical Theory of Communication.** Urbana: University of Illinois Press, 1964.
- [SMI 88] SMITHSON, M. **Ignorance and Uncertainty: Emerging Paradigms.** New York: Springer-Verlag, 1988.
- [STA 97] STAFF, Christopher. **HyperContext: Using Context in Adaptive Hypertext.** Disponível por WWW em <http://www.cs.um.edu.mt/~cstaff/HCTBrazil97/HCT97.html> (11 dez. 1997).
- [STO 89] STOTTS, P. David; FURUTA, Richard. Petri-Net based hypertext: Document structure with browsing semantics. **ACM Transactions on Information Systems**, New York, v.7 n.1, 1989. p. 3-29.
- [STO 90] STOTTS, P. David; FURUTA, Richard. The Trellis hypertext reference model. In: **HYPERTEXT STANDARDIZATION WORKSHOP, 1990**, Gaithersburg. **Proceedings...** Gaithersburg: NIST Special Publication, 1990. v.500-178.
- [THO 96] THOMAS, C. G.; FISCHER, G. Using agents to improve the usability and usefulness of the World-Wide-Web. In: **UM, INTERNATIONAL CONFERENCE ON USER MODELING, 5.**, 1996, Kailua-Kona. **Proceedings...** Kailua-Kona: User Modelling, 1996. p.5-12.
- [VAS 96] VASSILEVA, J. A task-centered approach for user modelling in a hypermedia office documentation system. **User Modeling and User-Adapted Interaction. Special issue on adaptive hypertext and hypermedia**, Dordrecht, v.6, n.2-3, p.87-129, 1996.
- [WAT 96] WATERWORTH, J. A. A pattern of islands: exploring public information space in a private vehicle. In: BRUSILOVSKY, P.; KOMMERS, P.; STREITZ, N. (Eds.). **Multimedia, Hypermedia and Virtual Reality.** Berlin: Springer-Verlag. (p. 12-30. Lecture Notes in computer Science, v.1077).
- [ZAK 96] ZAKON, R. H. Hobbes' Internet Timeline v.2.3, 1996. Disponível por WWW em <http://info.isoc.org/guest/zakon/Internet> (3 ago. 1997).



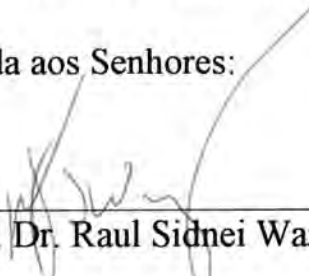
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

"Modelos Proativos para Hipermídia Adaptativa"


por

Luiz Antônio Moro Palazzo

Tese apresentada aos Senhores:



Prof. Dr. Raul Sidnei Wazlawick (UFSC)



Prof. Dr. Flávio Moreira de Oliveira (PUCRS)

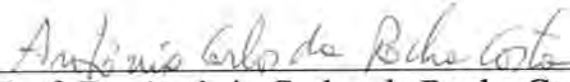


Profa. Dra. Rosa Maria Viccari

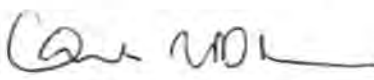
Vista e permitida a impressão.
Porto Alegre, 23 / 02 / 2000

in memoriam

Prof. Dr. José Mauro Volkmer de Castilho,
Orientador.



Prof. Dr. Antônio Carlos da Rocha Costa (UCPel),
Co-orientador.



Profa. Maria Dal Sasso Freitas,
Coordenadora do Programa de Pós-Graduação
em Computação - PPGC
Instituto de Informática - UFRGS